

Міністерство освіти і науки України

Український державний університет науки і технологій

Факультет Комп'ютерні технології та системи  
Кафедра Комп'ютерні інформаційні технології

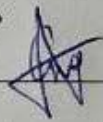
## Пояснювальна записка

до кваліфікаційної роботи  
магістра

на тему: «Дослідження методів моделювання часових послідовностей з  
нерівномірним кроком»

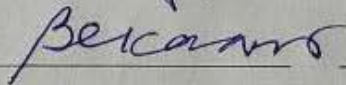
за освітньою програмою **12 Інженерія програмного забезпечення**  
зі спеціальності: **121 Інженерія програмного забезпечення**

Виконав: студент групи «ПЗ2221»



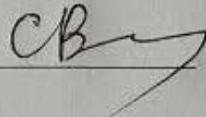
Андрій СТАРИНА

Керівник:



Владислав СКАЛОЗУБ

Нормоконтролер:



Світлана ВОЛКОВА

Засвідчую, що у цій роботі немає запозичень з  
праць інших авторів без відповідних посилань  
Студент

  
(Підпис)

Дніпро – 2024 рік

Ministry of Education and Science of Ukraine

Ukrainian State University of Science and Technologies

Faculty Computer technologies and systems  
Department Computer information technology

**Explanatory Note**  
to Master's Thesis

on the topic: «Research of methods for modeling time sequences with an uneven step»

according to educational curriculum **12 software engineering**  
in the speciality: **121 software engineering**

Done by the student of the group PZ2221:

\_\_\_\_\_  
Andrii STARYNA

Scientific Supervisor:

\_\_\_\_\_  
Vladyslav SKALOZUB

Normative controller:

\_\_\_\_\_  
Svitlana VOLKOVA

Дніпровський інститут інфраструктури та транспорту

Факультет Комп'ютерні технології і системи кафедра Комп'ютерні інформаційні технології

Спеціальність Інженерія програмного забезпечення

«ЗАТВЕРДЖУЮ»

Завідувач кафедри

\_\_\_\_\_ ±  
(підпис)

« \_\_\_\_\_ » \_\_\_\_\_ 2022 р.

### **ЗАВДАННЯ**

до дипломного проекту на здобуття ОС Магістр

(освітньо-кваліфікаційний рівень)

студента групи ПЗ2221

(номер групи)

Андрій СТАРИНА

(ПІБ)

1 Тема дипломного проекту: Дослідження методів моделювання часових послідовностей з нерівномірним кроком

затверджена наказом по університету від «» листопада 2020 р. № 8.

2 Термін подання студентом закінченого проекту «12» січня 2024 р.

3 Вихідні дані до дипломного проекту

послідовності спостережень недетермінованих процесів кланічного моніторингу хворих, отриманих з нерівномірним кроком спостережень; методи та процедури моделювання недетермінованих послідовностей, квантильні алгоритми; попередні результати сепарабельного моделювання часових послідовностей з нерівномірним кроком

4 Зміст пояснювальної записки (перелік питань до розробки) аналіз та дослідження ефективності методів та процедур моделювання часових послідовностей з перемінним кроком, процедури прогнозування, мульти-нечітка реляційно-сепарабельна модель аналізу та моделювання часових послідовностей з нерівномірним кроком, дослідження порівняльної ефективності алгоритмів моделювання з перемінним кроком, розробка програмних інструментальних засобів, проведення експериментів із порівняльного аналізу ефективності алгоритмів моделювання з перемінним кроком, економічний аналіз програмного забезпечення, технічна документація на розроблені програмні засоби.

5 Перелік демонстраційного матеріалу презентація завдання дослідження методів та процедур моделювання часових послідовностей з нерівномірним кроком, мульти-нечітка реляційно-сепарабельна модель та алгоритми її реалізації, процедури порівняльного аналізу алгоритмів моделювання, результати та висновки щодо ефективності алгоритмів аналізу та моделювання з перемінним кроком, розроблений програмний комплекс та інструментальні

засоби аналізу, відео-демонстрація роботи розробленого програмного комплексу

6. Консультанти (з назвами розділів):

Розділ	Консультант	Підпис, дата	
		завдання видав	завдання прийняв
Кошторис на розробку ПЗ	Професор Скалозуб В.В.		

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва розділів дипломної роботи	Термін виконання розділів роботи	Примітка
1	Вступ	01.09.2023–10.09.2023	
2	Аналіз сучасного стану дослідження проблеми моделювання часових послідовностей з нерівномірним кроком за науковими джерелами	11.09.2023–22.09.2023	
3	Аналіз засобів та сучасного стану програмно-апаратного забезпечення, яке потребує вдосконалення для вирішення завдань дослідження	23.09.2023–03.10.2023	
4	Постановка задачі дослідження, складання технічного завдання	04.10.2023–11.10.2023	30%
5	Техніко-економічні показники	12.10.2023–19.10.2023	
6	Розробка моделей, алгоритмів та інструментальних засобів дослідження	20.10.2023–25.10.2023	
7	Проведення досліджень	26.10.2023–06.11.2023	60%
8	Оформлення тез доповідей	07.11.2023–10.11.2023	
9	Підготовка статті у фаховий журнал	10.11.2023–12.11.2023	
10	Оформлення пояснювальної записки	13.11.2023–30.11.2023	
11	Розробка демонстраційних матеріалів	01.12.2023–12.01.2024	100%

Дата видачі завдання «\_\_» \_\_\_\_\_ 2022 р.

Керівник дипломного проекту \_\_\_\_\_ /Владислав СКАЛОЗУБ /  
(підпис) (ПІБ)

Завдання прийняв до виконання \_\_\_\_\_ / Андрій СТАРИНА /  
(підпис) (ПІБ)

## РЕФЕРАТ

Об'єктом дослідження є процеси моделювання та дослідження параметрів часових послідовностей, отриманих з нерівномірним кроком, які являються, зокрема, часовими рядами процесів моніторингу станів хворих.

Предметом дослідження є математичні моделі, алгоритми та програмні процедури щодо аналізу та прогнозування нерівномірних у часі недетермінованих числових послідовностей ,

Метою поданої роботи є підвищення ефективності та точності методів та алгоритмів аналізу та прогнозування характеристик недетермінованих послідовностей спостережень з нерівномірним кроком.

Методи дослідження: методи аналізу числових послідовностей, математичної статистики, методи порівняльного аналізу, комп'ютерне моделювання, процедури прогнозування, методи проектування та створення програмного забезпечення.

Результати та їх новизна: виконано порівняльний аналіз чисельної ефективності алгоритмів моделювання та прогнозування характеристик числових послідовностей з перемінним кроком спостережень, розроблено мульти-нечітку реляційно-сепарабельну модель прогнозування послідовностей з перемінним кроком, удосконалено процедуру формування комбінованих алгоритмів моделювання досліджуваних процесів.

Розрахунково-пояснювальна записка складається зі вступу, 4 розділів, висновків, бібліографічного списку та додатки

Вступ – містить сутність роботи та актуальність досліджень та розробок (3 сторінки).

Перший розділ – опис результатів аналізу сучасного стану досліджень ефективності методів та процедур моделювання часових послідовностей з перемінним кроком, за науковими літературними джерелами (24 сторінок).

Другий розділ – представлено сутність та відмінність досліджень щодо моделей та алгоритмів аналізу та прогнозування досліджуваних процесів, а також структуру нових та удосконалених процедур аналізу нерівномірних у часі числових послідовностей. (21 сторінок).

Третій розділ – представляє структуру, процес проектування і розробки інструментального забезпечення, призначеного для проведення дослідження (9 сторінок).

Четвертий розділ – дослідження властивостей алгоритмів моделювання та процедур аналізу та прогнозування. (8 сторінок).

Висновки складаються з 3 сторінок;

Список літератури – включає в себе список використаної літератури. Складає 5 сторінок;

Додатки – технічне завдання і робочий проект. Таблиць – 6, рисунків – 10, бібліографія – 65.

Ключові слова: часова послідовність, нерівномірний крок спостережень, мульти-нечітка реляційно-сепарабельна модель, ефективність комбінованих алгоритмів, прогнозування, програмне забезпечення.

## ЗМІСТ

Вступ.....	9
1. Постановка завдань дослідження, аналіз методів та засобів моделювання часових послідовностей та їх прогнозування.....	12
1.1. Аналіз предметної сфери моделювання недетермінованих процесів ...	12
1.1.1. Опис особливостей завдань моделюванн. Актуальність дослідження	13
1.1.2. Огляд результатів основних наукових досліджень .....	18
1.1.3. Огляд засобів програмних аналогів .....	25
1.2. Призначення та сфера застосування результатів досліджень .....	29
1.3. Постановка завдань дослідження .....	32
Висновки до розділу 1 .....	35
2. Обґрунтування та методика досліджень алгоритмів із моделювання та прогнозування недетермінованих процесів з перемінним кроком .....	36
2.1. Дослідження ефективності алгоритмів із моделювання та прогнозування числових послідовностей з перемінним кроком.....	36
2.1.1. Процедури аналізу та прогнозування характеристик на основі квантильних алогоритмів .....	37
2.1.2. Аналіз напрямків удосконалення алгоритмів на основі нечіткого моделювання процесів, мульти-нечіткі комбіновані алгоритми. процесів .....	40
2.1.3. Реляційно-сепарабельна модель моделювання та процедури дослідження чисельної ефективності та точності алгоритмів .....	42
2.1.4. Мульти-нечітка квантильна модель недетермінованих послідовностей з перемінним кроком.....	49
2.2. Дослідження функціональної ефективності алгоритмів моделювання недетермінованих послідовностей з перемінним кроком.....	51
2.3. Особливості структури процесів моделювання числових послідовностей з перемінним кроком.....	52
Висновок до розділу 2 .....	56

3. Розробка структури та інструментальних засобів для дослідження алгоритмів моделювання числових послідовностей з перемінним кроком....	57
3.1. Формалізація завдань дослідження алгоритмів моделювання.....	57
3.2. Розробка базової архітектури інструментальних засобів .....	58
3.3. Внутрішнє проектування інструментальних засобів .....	59
3.3.1. Обґрунтування та вибір мови програмування .....	59
3.3.2. Розробка архітектури ієрархії та взаємодії класів системи інструментальних засобів.....	61
3.4. Формування інтерфейсу користувача програмного комплексу.....	62
3.5. Тестування та налагодження програми .....	63
3.5.1. Аналіз та застосування методів тестування та налагодження .....	63
Висновки до розділу 3 .....	65
4. Порівняльні дослідження ефективності алгоритмів класифікації та моделювання числових послідовностей з нерівномірним кроком .....	66
4.1. Постановка завдань та підготовка даних експерименту.....	66
4.1.1. Опис структури засобів програмно-апаратного середовища.....	68
4.1.2. Алгоритми та процедури методу моделювання нерівномірних часових послідовностей.....	69
4.1.3. Опис методики відбору даних для експериментальних досліджень алгоритмів.....	70
4.2. Процедура та результати проведення експериментальних досліджень алгоритмів моделювання нерівномірних часових послідовностей .....	71
Висновки до розділу 4 .....	73
5. Висновки .....	74
Список використаної літератури .....	77
Додатки.....	82

## ВСТУП

**Актуальність роботи.** Комп'ютерне моделювання, аналіз і прогнозування характеристик складних процесів і систем являються необхідною складовою досліджень, проектування, розробок технологічних процесів. Моделі, методи та програмні засоби щодо ефективного та достовірного застосування процедур комп'ютерне моделювання на основі числових рядів спостережень являються актуальними. Складність процесів, які досліджуються являється одним із головних факторів, що визначають можливість застосування певних процедур аналізу та прогнозування, а також визначають ефективність програмних систем,

В багатьох випадках аналізу технологічних процесів при аналізі та моделюванні складних систем як вихідні дані можливо отримати лише часові послідовності характеристик процесів, часові ряди (ЧР). Натепер моделі та методи аналізу та моделювання ЧР з перемінним кроком спостережень є не достатньо дослідженими. А завдання щодо розвитку методів та засобів моделювання нерівномірних у часі часових послідовностей досліджені не в повній мірі.

Розвиток досліджень і розробок засобів програмного забезпечення щодо моделей, алгоритмів і програмних засобів, призначених для аналізу нерівномірних у часі послідовностей спостережень, має науковий і практичний інтерес. Таким чином, тема досліджень представленої роботи являється натепер актуальною .

**Методи аналізу та дослідження,** що використовувалися в роботі, складаються з методів системного аналізу, методи аналізу даних, методів аналізу часових рядів, методи математичного моделювання та математичної статистики та порівняльного аналізу, методів нечіткого моделювання, а також комп'ютерного моделювання. В роботі використані методи та технології щодо проектування та створення програмних систем, методи алгоритмізації; експериментального дослідження програмного забезпечення.

**Об'єктом дослідження** є процеси моделювання та дослідження параметрів часових послідовностей отриманих з нерівномірним кроком, які являються, зокрема, часовими рядами процесів моніторингу станів хворих

**Предметом дослідження** являються математичні моделі, алгоритми та програмні процедури щодо аналізу та прогнозування нерівномірних у часі недетермінованих числових послідовностей.

**Мета роботи** це підвищення ефективності та точності методів і алгоритмів аналізу та прогнозування характеристик недетермінованих послідовностей спостережень з нерівномірним кроком.

**Завдання.** Завдання дослідження полягали у наступному:

- розробити алгоритмічні та програмні засоби та виконати порівняльний аналіз чисельної ефективності алгоритмів моделювання нерівномірних у часі недетермінованих числових послідовностей даних спостережень.
- розробити алгоритмічні та програмні засоби та виконати прогнозування .... часових послідовностей,
- розробити інструментальне програмне середовище, яке забезпечує дослідження алгоритмів моделювання нерівномірних у часі недетермінованих числових послідовностей;
- виконати порівняльний аналіз алгоритмів моделювання нерівномірних у часі послідовностей щодо точності та чисельної ефективності;
- отримати рекомендації стосовно застосування процедур моделювання недетермінованих послідовностей з нерівномірним кроком спостережень.

**Методи дослідження** складаються з методів системного аналізу, методів аналізу даних, методів аналізу часових рядів, методів математичного моделювання та математичної статистики та порівняльного аналізу, методів нечіткого моделювання, а також комп'ютерного моделювання. В роботі використані методи та технології щодо проєктування та створення програмних систем, методи алгоритмізації; експериментального дослідження програмного забезпечення.

**Наукова новизна** результатів роботи визначається наступним

- виконано порівняльний аналіз чисельної ефективності алгоритмів моделювання та прогнозування характеристик числових послідовностей з перемінним кроком спостережень,
- розроблено мульти-нечітку реліційно-сепарабельну модель прогнозування послідовностей з перемінним кроком,
- удосконалено процедуру формування комбінованих алгоритмів моделювання досліджуваних процесів, що забезпечує коректність результатів комп'ютерного моделювання та підвищення точності показників процесів.

**Практичне значення** мають результати проведеного аналізу стану щодо застосування моделей та алгоритмів моделювання та прогнозування часових послідовностей спостережень, що мають нерівномірний крок спостережень, результати порівняльного аналізу чисельної ефективності алгоритмів моделювання та прогнозування характеристик числових послідовностей з перемінним кроком спостережень, результати застосування процедур формування комбінованих алгоритмів моделювання досліджуваних процесів, що забезпечує коректність результатів комп'ютерного моделювання та підвищення точності показників процесів, програмні засоби для моделювання та прогнозування характеристик числових послідовностей з перемінним кроком спостережень,.

**Апробація результатів дослідження.** Результати дослідницької роботи доповідались на семінарах кафедри КІТ 30.11.2022 р. та 11.01.2024 р. Матеріали роботи доповідалися на 16 та 17 міжнародних науково-практичних конференціях «Сучасні інформаційні та комунікаційні технології на транспорті, в промисловості та освіті» (14 – 15 грудня 2022 р. та 13-14 грудня 2023 р..) м. Дніпро, Український державний університет науки і технологій.

# 1. ПОСТАНОВКА ЗАВДАНЬ ДОСЛІДЖЕННЯ, АНАЛІЗ МЕТОДІВ ТА ЗАСОБІВ МОДЕЛЮВАННЯ ЧАСОВИХ ПОСЛІДОВНОСТЕЙ ТА ЇХ ПРОГНОЗУВАННЯ

## 1.1. Аналіз предметної сфери моделювання недетермінованих процесів.

Дипломна робота присвячена дослідженню та реалізації програмних засобів для виконання актуальних завдань щодо моделювання і аналізу даних часових послідовностей моніторингу процесів з нерівномірними інтервалами вибірки та іншими характеристиками, які потребують комп'ютерне моделювання. Натепер застосування комп'ютерного моделювання, що забезпечує аналіз і прогнозування характеристик складних процесів і систем, являються необхідною складовою наукових та прикладних досліджень, проектування, розробок технологічних процесів. Складність процесів, які досліджуються являється одним із головних факторів, що визначають можливість застосування певних процедур аналізу та прогнозування, а також визначають ефективність програмних систем. При моніторингу стану інформаційних та виробничих систем, в завданнях аналізу станів технологічних процесів, при аналізі та моделюванні складних систем тощо, як достовірні та сталі вихідні дані можливо отримати лише величини характеристик процесів, упорядкованих у часові послідовності, часові ряди (ЧР). Моделі, методи та програмні засоби щодо ефективного та достовірного застосування процедур комп'ютерне моделювання на основі таких числових рядів спостережень являються актуальними. При цьому на практиці сають місце процеси, в яких характеристики ЧР вимірюються з нерівномірним кроком. Такі ЧР характерні для багатьох процесів моніторингу станів складних технічних, інформаційних процесів, а також для важливих і відповідальних процесів клінічного моніторингу лікування хворих, процесів реабілітації.

Незважаючи на важливість процесів з перемінним кроком, огляд літературних джерел показав, що математичні моделі та методи аналізу та моделювання ЧР з перемінним кроком спостережень є не достатньо

дослідженими. Також відзначається, що завдання із розвитку методів та програмних засобів аналізу і прогнозування нерівномірних у часі часових послідовностей досліджені не в повній мірі. Назване свідчить про своєчасність і актуальність тематики дипломної роботи.

#### 1.1.1. Опис особливостей завдань моделювання. Актуальність дослідження.

В дипломній роботі вирішуються завдання щодо удосконалення математичних та інформаційних моделей, алгоритмів та програмних процедур, призначених для дослідження недетермінованих часових послідовностей, які представляють складні процеси що характеризуються перемінним кроком спостережень у ЧР. Розвиток досліджень і розробок засобів програмного забезпечення щодо моделей, алгоритмів і програмних засобів, призначених для аналізу нерівномірних у часі послідовностей спостережень, має науковий і практичний інтерес. В змістовному сенсі саме існування нерівномірного інтервалу між спостереженнями в ЧР представляє глобальну особливість особливостей завдань моделювання, робить дослідження та розробки роботи на тепер актуальними.

Відзначимо, що задача з аналізу, прогнозування та оперативного планування значень параметрів економічних, технологічних і інформаційних процесів складних систем, до яких відносяться і процеси залізничного транспорту, і загального та клінічного моніторингу, все більше актуальні в умовах змін зовнішнього середовища. Вони виникають на різних рівнях прийняття рішень, їх достовірне та вчасне вирішення забезпечує ефективне функціонування багатьох різноманітних систем. При цьому має місце різноманітність видів методів для представлення ЧР. На рис.1.1 подана класифікація типів ЧР, відповідно форм утворення та представлення первинних даних.

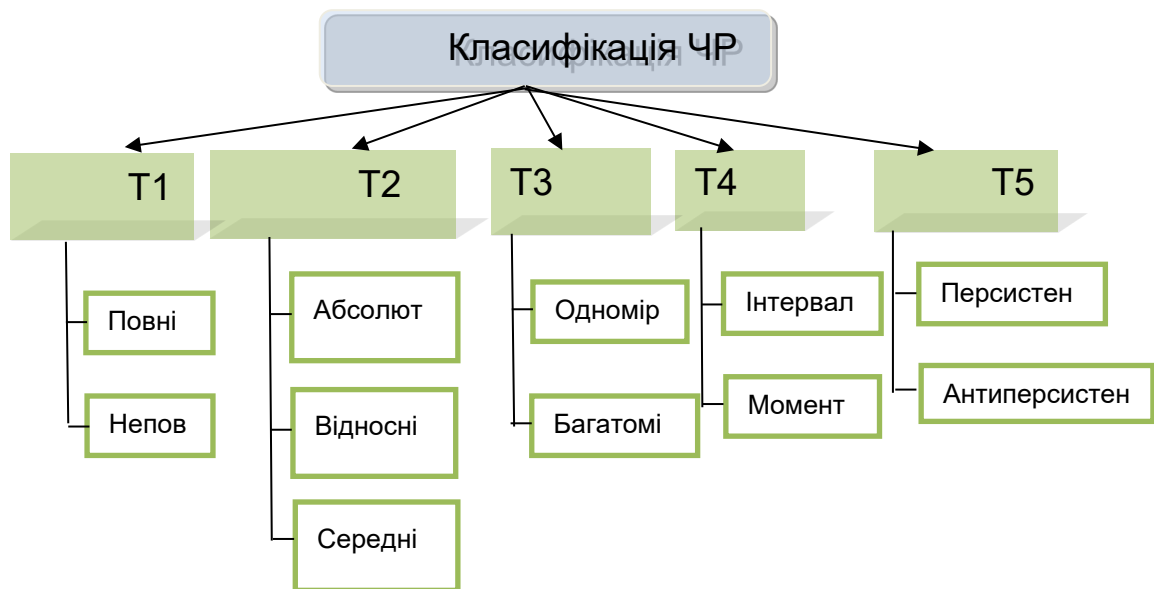


Рисунок 1.1. Класифікація .видів часових рядів процесів функціонування складних систем

Рис. 1.1 характеризує широкий спектр типів ЧР і способів представлення. Необхідно зазначити, що часові послідовності процесів далі досліджують без урахування та аналізу впливу зовнішніх факторів. Тому цей аналіз є неповним, потребує застосування процесів інтерпретації. Указані способи утворення та вимоги до параметрів спостережень ЧР, які в переважній більшості передбачають рівномірний інтервал, повинні чітко виконуватися. Не дотримання цього може приводити до ризиків помилкового результату, до високої ймовірності отримати хибний тобто не адекватний даним спостережень результат. Відзначимо що класифікація рис. 1.1 не указує на особливості категорій ЧР з нерівномірним кроком між спостереженнями.

В якості відомого та такого що досить часто використовується для випробувань методів та алгоритмів аналізу та моделювання ЧР покажимо процес ТАІFЕХ, вихідна послідовність значень та графік якого демонструє рис. 1.2. В наступному розділі нами буде приведений досить простий квантильний алгоритм, який був створений та випробуваний для процесів ТАІFЕХ, а також подібних.

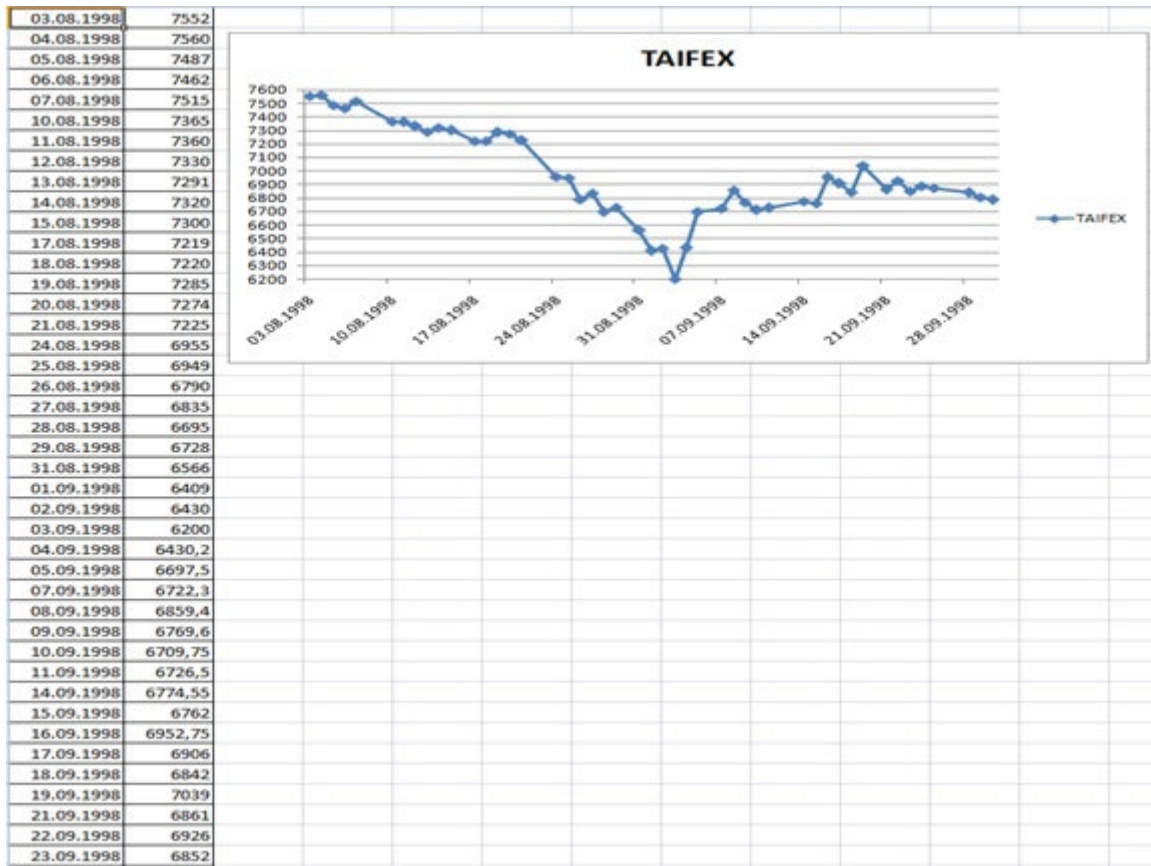


Рисунок 1.2 Представлення даних варіанту процесів TAIFEX

Як зазначено вище, існує широкий спектр процесів моніторингу, які принципово мають нерівномірний крок між спостереженнями, причому спроби не враховувати такі особливості суттєво спотворюють властивості процесів. На рис. 1.3, рис. 1.4 приведені приклади представлення фрагментів процесів клінічного моніторингу, які мають нерівномірний крок між вимірюваннями. Недетермінованість та нерівномірність кроку ЧР спостережень, а також мале число даних, суттєво ускладнюють і навіть частково неможливають застосування до них стандартних моделей та процедур класичного статистичного аналізу. Необхідно відзначити, що періоди моделей процесів моніторингу реабілітації надзвичайно нерівномірні, вони практично унікальні для кожного пацієнта, особливі та відмінні для різних хворих. Попередньо результати моделювання завдань моніторингу реабілітаційних процесів з нерівномірними у часі подіями показали необхідність подальших докладніших розробок або модифікації вже відомих алгоритмів.



Рисунок 1.3. Фрагмент первинних вихідних даних спостережень процесу клінічного моніторингу пацієнта П45

(1.17)



Рисунок 1.4. Фрагмент первинних даних спостережень процесу клінічного моніторингу пацієнта П49

Головним завданням та задачею аналізу ЧР подібних до рис. 1.3, рис. 1.4, формально являється розрахунок значення величини наступного кроку в процесах моніторингу (визначенні оцінок занчень довжини кроку, а також очікуваної величини контрольованого показника).

В дипломній роботі нами досліджено актуальні завдання щодо підвищення ефективності процедур моделювання і аналізу даних часових послідовностей моніторингу процесів з нерівномірними інтервалами вибірки та іншими характеристиками. При цьому відзначається що в складних системах деякі параметри станів, контрольовані характеристики процесів тощо, ожуть мати значну ступінь невизначеності. На практиці процедури моніторингу таких процесів часто дають змогу отримати лише короткі часові послідовності даних з нерівномірною у часі вибіркою приводять до того, що завдання аналізу суттєво ускладнюється.

Реалізація завдань аналізу і прогнозування зазначених часових рядів (послідовностей даних спостережень) виконується на основі запропонованої у роботах [1, 2] сепарабельної математичної моделі (СПМ) недетермінованих процесів. СПМ відрізняється окремим формуванням вибірки величин показників процесу та інтервалів між спостереженнями. В [1, 2] будується сепарабельна модель, яка призначена для дослідження процесів клінічного моніторингу стану хворих на діабет. Така модель застосовується з метою визначення оцінок величин нового періоду до виникнення стану/подій, які відповідають встановленим вимогам. В моделі СПМ нерівномірні інтервали вибірок виділяються в окрему складову, а також розглядаються як одна із компонент вектору даних процесу моніторингу. При моделюванні часової складової враховуються лише послідовності величин інтервалів. Разом з цим окремо моделюються інші характеристики процесів, які узгоджуються за порядком їх розміщення у даних НЧПД. Отримані результати моделювання всіх характеристик у підсумку «накладаються» на результати моделювання часових інтервалів. СПМ модель процесів моніторингу у цілому визначає як значення нового моменту виникнення чергової події спостережуваного процесу, так і відповідні кожному моменту його характеристики.

#### 1.1.2. Огляд результатів основних наукових досліджень

Розглянемо основні положення, моделі та методи і алгоритми, які використовувалися для моделювання та дослідження НЧПД. Тематика

дипломної роботи відповідає напряму досліджень щодо моделювання та прогнозування недетермінованих процесів у формі ЧР, які наведені в публікаціях

1. Скалозуб В.В., Білий Б.Б., Галабут О.О., Мурашов О.В. (2020). Методи інтелектуального моделювання процесів з перемінним інтервалом спостережень та конструктивного упорядкування «з вагою». Системные технологии. Региональный межвузовский сборник научных работ. Выпуск 3 (128). 127-143.
2. Скалозуб В.В., Мурашов О.В. (2021) Моделювання даних процесів моніторингу при нерівномірних і нечітких інтервалах спостережень. «Системні технології». Выпуск 4 (135). 135-148
3. Аналіз та прогнозування нерегулярних числових послідовностей з перемінним та нечітким інтервалом спостережень. // Скалозуб В. В., Мурашов О.В. // VI Міжнародна науково-технічна конференція «комп'ютерне моделювання та оптимізація складних систем», М Дніпро, УДХТУ, 4-6 листопаду. 2020, С. 154-156..

Також в поданих дослідженнях використані певні результати інших наукових статей, які наведені далі.

В роботі [7] було удосконалено науково-методичний підхід до моделювання процесів НР з перемінним кроком спостережень. При цьому було зазначено, що головною змістовною передумовою, певним аксіоматичним твердженням, що призване забезпечити можливість існування сепарабельної моделі складних процесів, являється системна єдність всіх властивостей досліджуваного процесу, поданих через високу складність зв'язків окремих характеристик у вигляді часового ряду. З цього витікає що перемінні інтервали спостережень є окремою ознакою процесу, його властивістю поряд з іншими характеристиками. Виходячи з наведеного будемо представляти сепарабельну модель (СПМ) процесу (моніторингу

НЧПД  $F(t)$  та інших) у наступному вигляді:

$$SpM(t) = SpMt(k) \cup \{SpMi(SpMt(k))\}, i=1, 2, \dots, q \quad (1.1)$$

де позначено

$t$  – координата часу,

$i$  – індекси вектору характеристик процесу моніторингу,

$k$  – порядковий номер рівня вибірки даних  $SpM(k)$ .

СПМ (1.1) утворюється шляхом поєднання моделей окремих характеристик  $F(t)$ , розрахованих для моментів, визначених на основі нечіткої моделі часу процесу  $SpMt(k)$ . При цьому моделі окремих характеристик процесу  $\{SpMi(SpMt(k))\}$  поєднуються та узгоджуються між собою параметром  $k$  – порядковим номером рівня моделі вибірки даних  $SpM(k)$ .

Рівняння (1.1) є визначенням сепарабельних моделей багатопараметричних недетермінованих процесів.

Для контролю достовірності та точності аналізу та прогнозування НЧР процесів клінічного моніторингу реабілітації на основі застосування СПМ (1.1) за допомогою удосконаленого алгоритму був застосований підхід, що передбачав розмноження вихідної нечіткої часової послідовності методом «бутстреп». Така генерована модель дозволяла отримати достатню кількість вхідних даних з математичними властивостями, які відповідають вхідній НЧП процесів моніторингу. Для демонстрації можливостей нижче подано результати застосування моделей СПМ для аналізу та прогнозування процесів клінічного моніторингу хворих (персона П45, рис. 1.3)

На рис 1.5 приведені упорядковані за появою у процесі значення рівнів показника вмісту цукру в крові, а на рис. 1.6 приведені відповідні за порядком часові інтервали між спостереженнями.

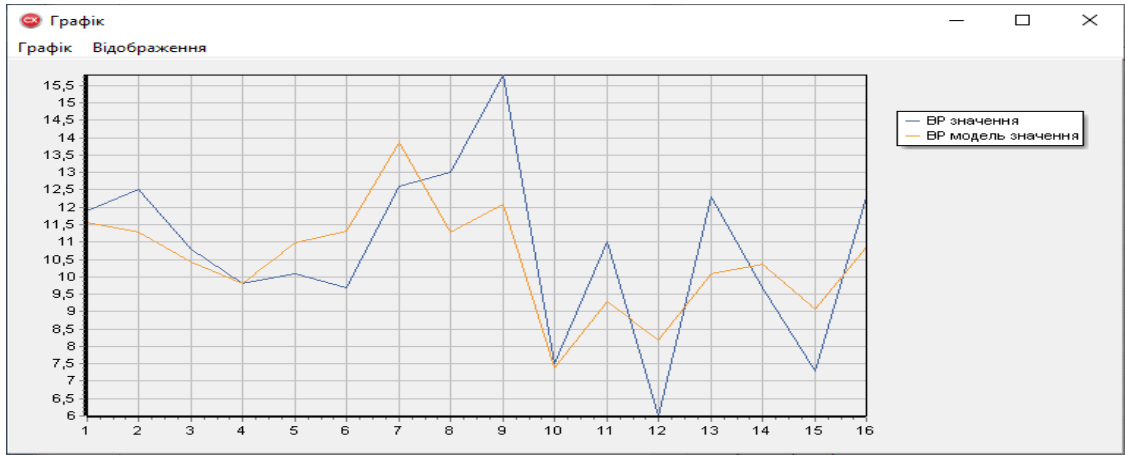


Рисунок 1.5 Результати прогнозування кількісного показника процесів моніторингу пацієнта П45

На рис. 1.5 і рис. 1.6 світлими лініями відображені результати моделювання узагальнених та узгоджених за номерами значень оцінок послідовностей параметрів моніторингу хворого П1945. Рис. 1.7 представляє підсумок, отриманий у відповідності до СПМ (1.1)

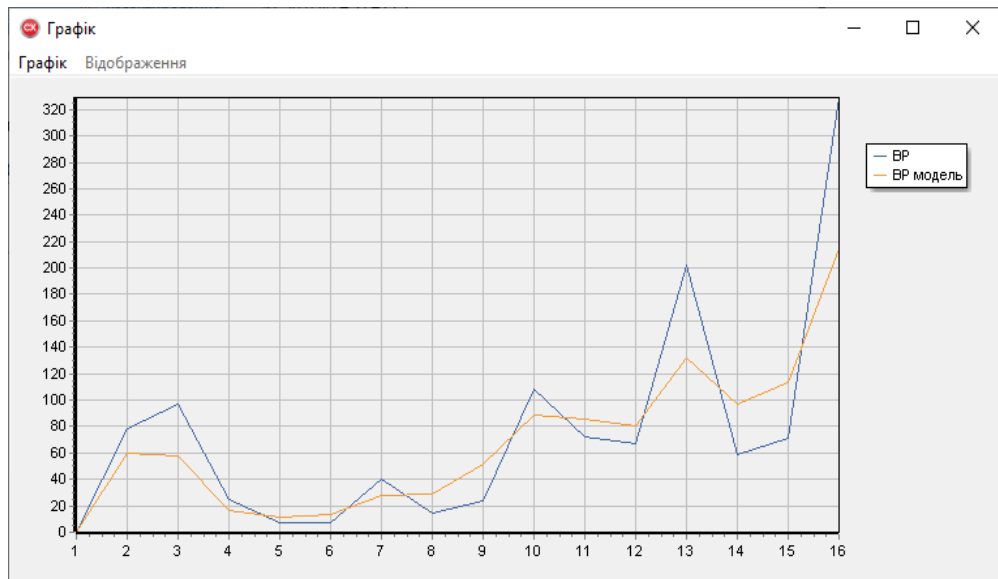


Рисунок 1.6 Результати моделювання часових інтервалів для процесів моніторингу реабілітації П45

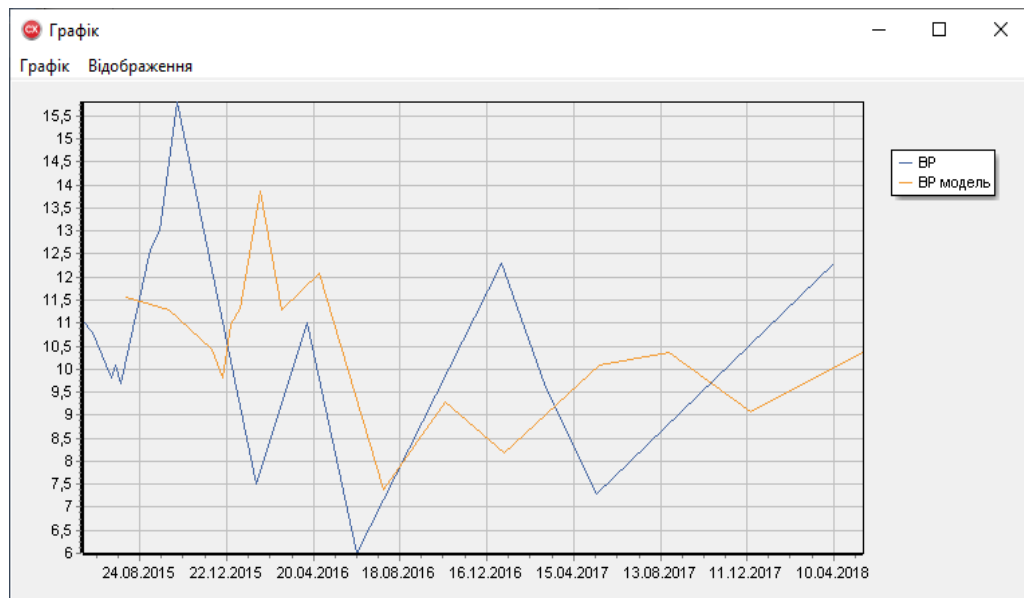


Рисунок 1.7 Графік сепарабельної моделі процесу моніторингу П45

Рис. 1.7 демонструє застосування моделі СПМ до даних процесу моніторингу П45. При цьому оцінки рівнів показника «ППП» та інтервалу до наступної події визначалися стандартними процедурами квантильного алгоритму, що описаний нижче. На вісі абсцис рис. 1.5 та рис. 1.6 відкладені номери послідовності спостережень, за якими узгоджуються в СПМ моделі окремі складові числової послідовності кроків спостережень.

Для моделювання та дослідження моделей недетермінованих процесів моніторингу у формі НЧПД, в роботах [5] були застосовані удосконалені квантильні моделі процесів, які відповідали усім передумовам методології FTS [1, 2]. При цьому розглядають НЧР  $F(t)$ , а також  $R(t,t-1)$  - модель першого порядку  $F(t)$ . У разі визначення для  $F(t)$  нечітких множин  $F(t-1)$ ,  $F(t-2)$ , ...,  $F(t-n)$  має місце відношення n-го порядку

$$F(t-n), \dots, F(t-2), F(t-1) \rightarrow F(t)$$

Особливість цього методу прогнозування полягає у введення показника моделювання «Тренд» та параметру  $k$  (каппа), який дозволяє характеризувати очікуваний напрям тренду процесу. Показник «Тренд» визначає напрямок послідовності даних, який розраховується за (1.2) – (1.4)

$$\left( (y_{n-1} - y_{n-2}) - (y_{n-2} - y_{n-3}) \right) > k \quad (1.2)$$

$$\left( (y_{n-1} - y_{n-2}) - (y_{n-2} - y_{n-3}) \right) < k \quad (1.3)$$

$$\left( (y_{n-1} - y_{n-2}) - (y_{n-2} - y_{n-3}) \right) = k \quad (1.4)$$

де  $n$  – поточний рівень ітерації;  $y_n$  – значення величини показника НЧР;  $k$  – параметр «каппа». Коли «Тренд» відповідає умові (1.2), прогнозоване значення розраховується за формулою (1.5).

$$t_j = \frac{2}{\frac{0.25}{m_{j-1}} + \frac{1}{m_j} + \frac{0.75}{m_{j+1}}} \quad (1.5)$$

При виконанні умови (1.3) прогнозоване значення розраховується за (1.6).

$$t_j = \frac{2}{\frac{0.75}{m_{j-1}} + \frac{1}{m_j} + \frac{0.25}{m_{j+1}}} \quad (1.6)$$

За умови (1.4) прогнозоване значення розраховується відповідно (1.7).

$$t_j = \frac{2}{\frac{0.5}{m_{j-1}} + \frac{1}{m_j} + \frac{0.5}{m_{j+1}}} \quad (1.7)$$

де  $m_{j-1}$ ,  $m_j$  та  $m_{j+1}$  - середні точки інтервалів  $u_{j-1}$ ,  $u_j$  та  $u_{j+1}$  з відповідними лінгвістичними термами  $A_{j-1}$ ,  $A_j$  та  $A_{j+1}$ .

Особливість наведеного алгоритмів [2, 5] полягає у оцінюванні трендів процесів  $f(y)$ . В залежності від показників трендів виконується формування прогнозованих оцінок контрольованих величин. Для оцінювання розвитку процесів встановлюється чисельний параметр – константа «каппа», значення якого впливає на вибір напрямку «тренду» при моделюванні. Відповідно до [2, 5] значення параметру «каппа» встановлюється методом спроб та помилок. В роботі [5] параметр «каппа» був представлений діапазоном, рис. 1.8. Границі діапазону «каппа» розраховуються відповідно (1.8).

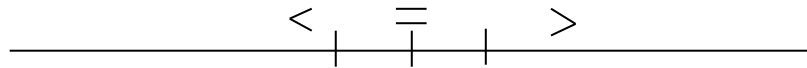


Рисунок 1.8 – Інтервальна модель параметру тренду «каппа»

$$k_1 = \text{kappa} - \Delta k, k_2 = \text{kappa} + \Delta k \quad (1.8)$$

де  $\Delta k$  – відсотковий параметр (наприклад 5%) від значення центру «каппа».

Також модифікація алгоритму (1.2) – (1.7) полягала розрахунку «тренду» відповідно (1.9) – (1.11).

$$((y_{n-1} - y_{n-2}) - (y_{n-2} - y_{n-3})) > k_2, \quad (1.9)$$

$$((y_{n-1} - y_{n-2}) - (y_{n-2} - y_{n-3})) < k_1, \quad (1.10)$$

$$((y_{n-1} - y_{n-2}) - (y_{n-2} - y_{n-3})) = [k_1, k_2], \quad (1.11)$$

де значення  $y_n$  – належить функції  $f(y)$ ;

$k_1$  – ліва межа параметру «каппа»;  $k_2$  – права межа параметру «каппа».

При тому прогнозі значення рівнів показника так. При виконанні умов (1.9) тобто рух вгору, прогнозне значення розраховується за (1.12).

$$t_n = \frac{2}{\frac{0.25}{y_{n-2}} + \frac{1}{y_{n-3}} + \frac{0.75}{y_{n-1}}}. \quad (1.12)$$

При виконанні умов (1.10) тобто рух вниз, прогнозне значення розраховується за формулою (1.13)

$$t_n = \frac{2}{\frac{0.75}{y_{n-2}} + \frac{1}{y_{n-3}} + \frac{0.25}{y_{n-1}}}. \quad (1.13)$$

Якщо «тренд» виконує умову (1.11) тобто рух на своєму рівні, прогнозне значення розраховується за (1.14)

$$t_n = \frac{2}{\frac{0.5}{y_{n-2}} + \frac{1}{y_{n-3}} + \frac{0.5}{y_{n-1}}}, \quad (1.14)$$

де  $n$  – поточний рівень ітерації;  $y_n$  – належить функції  $f(y)$ ;  $t_n$  – прогнозоване значення поточного рівня.

Відзначається що за даними чисельних експериментів була визначена висока швидкість алгоритму, але встановлене зменшення точності прогнозування при використанні рівнянь (1.12) – (1.14), якщо кількість кроків процесу вперед більше 3-х.

Крім наведених вище модифікації були досліджені можливості створення комбінованих алгоритмів, які поєднували можливості базового квантильного алгоритму (1.2) – (1.7) та запропонованих модифікації, тобто моделі (1.9) – (1.14). Величина параметру « $\alpha$ » підбирається на основі процедури мінімізації квадрату середньої похибки

$$Y_k = \alpha V_k + (1-\alpha)u_k \quad 0 \leq \alpha \leq 1 \quad (1.15)$$

на основі (1.5) – (1.7) для  $V_k$ , а також для  $U_f(t)$ , що відповідає алгоритму статті [7], з підбором параметру « $\alpha$ »

$$Y_k = \alpha V_k + (1-\alpha)U_f k \quad (1.16)$$

За схемою (1.16) результат формується відповідно  $V_k$ , для схеми (1.15), але в разі величин вхідного ЧР беруться значення (1.17)

$$z_k = \alpha x_k + (1-\alpha)m_k \quad (1.17)$$

В роботі були виконані експериментальні дослідження вдосконаленого квантильного методу, (ВКМ), представленого вище.

Аналіз зазначених вище досліджень алгоритмів аналізу і моделювання НЧПД дозволяє зробити наступні висновки

- п1) один із головних параметрів квантильного алгоритму «каппа», від значень якого суттєво залежить результат моделювання та прогнозування, являється поки що евристичним, вибір його значень не представлений, як і можливі засоби автоматизованого вибору,

- п2) для удосконалення та підвищення чисельної ефективності алгоритмів важливим напрямком розвитку являється дослідження схем (1.15) – (1.17), тобто формування комбінованих процесів, які поєднують різні схеми використання параметрів первинних процесів, поточних даних. .

Дослідження щодо можливостей удосконалення алгоритмів відповідно пунктів п1 та п2 виконані в дипломній роботі.

### 1.1.3. Огляд засобів програмних аналогів

Досліджуючи спектр програм аналізу НЧПД було встановлено, що на тепер в загальноприйнятих програмних комплексах моделювання та аналізу таких процесів стандартні засоби, які спираються на певні математичні теорії та достовірні результати, нами не виявлені. При тому результати огляду показали, що в програмних системах (наприклад Python) існують спеціалізовані бібліотеки, за допомогою яких потенційно можлива реалізація окремих завдань послідовностей НЧПД.

Разом з тим виявлено існування окремих спеціалізованих програм, які реалізують алгоритми виду (1.2) – (1.7). та (1.9) – (1.12). В них також частково реалізовані окремі завдання формування та аналізу сепарабельних моделей СПМ. При тому програми орієнтовані на саме завдання клінічного моніторингу, мають обмежену сферу застосування, їх використання для інших недетермінованих НЧПД процесів потребує додаткового програмування.

Приведемо приклад результатів моделювання на основі СПМ, щоб підтвердити працездатність методу. На рис. 1.9 приведені результати моделювання процесів TAIFEX модифікованим алгоритмом ВКМ, що

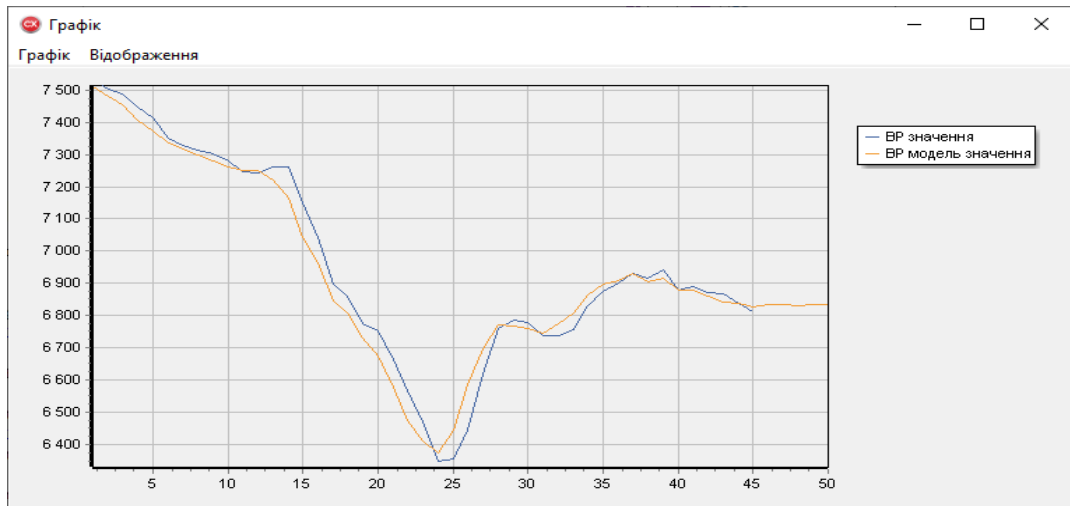


Рисунок 1.9 – Результати моделювання процесів ТАIFEX алгоритмом ВКМ при згладжуванні тренду за 3 рівнями

свідчить про досить його високі якості стосовно достовірності і точності.

В роботі [Ск-Мур] було виконано перевірку придатності та точності модифікацій комбінованих алгоритмів виду (1.15) для сепарабельних моделей, які відповідають прогнозуванню ТАIFEX [7]. При цьому дані ТАIFEX були трансформовані так, щоб відобразити задану нерівномірність інтервалів оцінювання процесів, позначимо як ТАIFEX-Т. Для створення інтервальної нерівномірності умовно вважалось, що дані надходять через періоди часу, які пропорційні величинам показників. На рис. 1.10 показана варіативність періодів (інтервалів) між спостереженнями, на рис. 1.11 масштабовані значення моделі для величин показників. На рис. 1.12 приведені результати застосування СПМ моделі ТАIFEX-Т. Указується, що кожна із складових моделі СПМ мала точність до 1,5%.

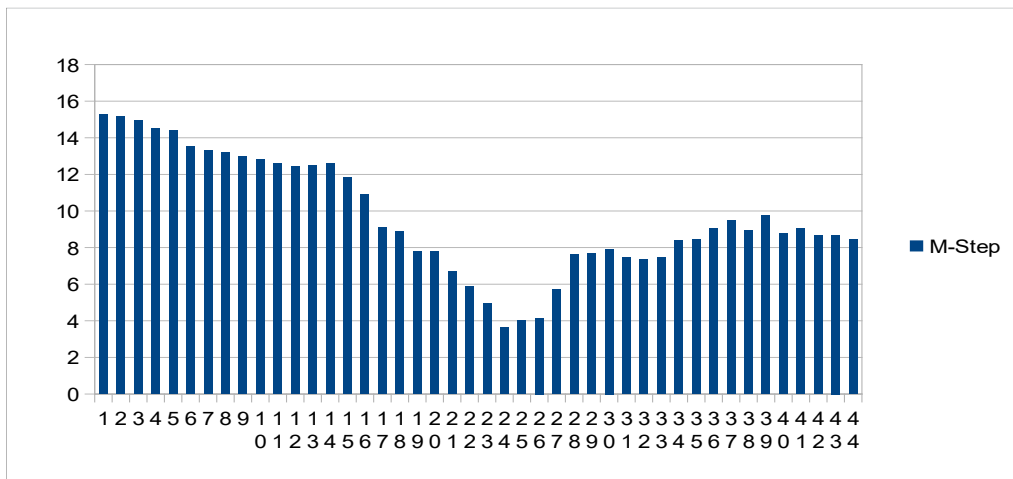


Рисунок 1.10 Результати моделювання послідовності інтервалів між даними процесу TAIEX-T (за номерами спостережень)

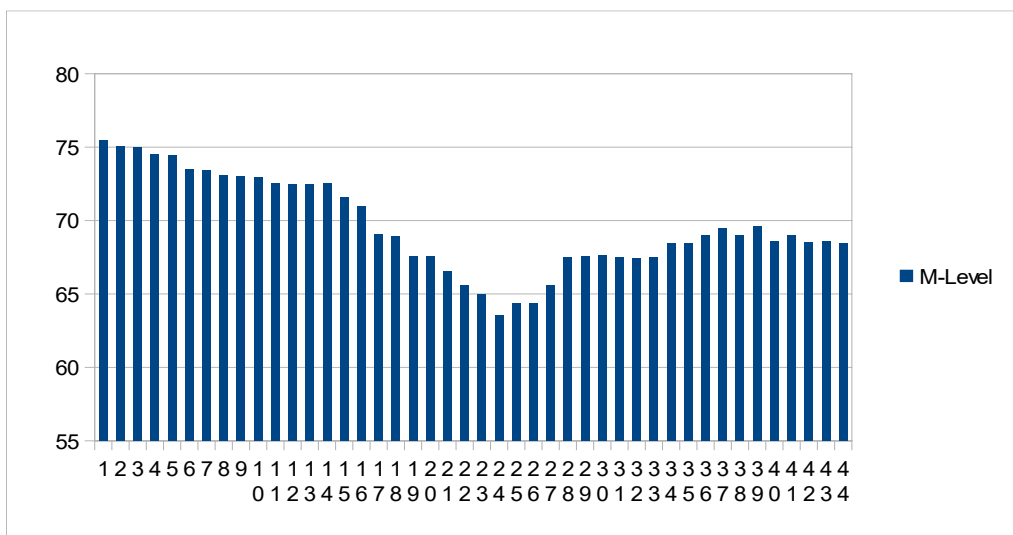


Рисунок 1.11 Результати моделювання послідовності рівнів процесу (за номерами спостережень)

В роботі [20] також були досліджені результати застосування СПМ для аналізу та прогнозування завдань клінічного моніторингу, рис. 1.3, рис. 1.4. При цьому отримано, що в цілому модель СПМ відтворювала процес клінічного моніторингу пацієнта при нерівномірних інтервалах контролю з похибкою до 5% , що можна вважати достатнім, стосовно забезпечення вірних встановлених вимог щодо оцінки стану хворого. Таким чином, сепарабельні моделі в цілому вірно відтворюють більшу частину кількісних рівнів вхідного ряду, що було забезпечене відповідно і значень

контрольованих характеристик, і часових компонентів. Також певною мірою отримані результати підтвердили можливості (для досліджуваного процесу

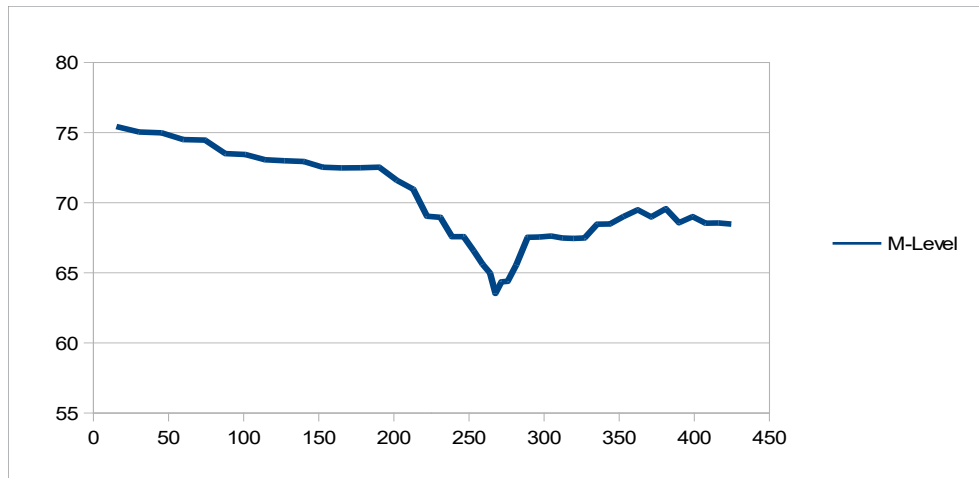


Рисунок 1.12 Графічне представлення процесу TAIFEX-T з перемінним інтервалом на основі сепарабельної моделі СПМ

моніторингу) прийнятих властивостей загальної сепарабельної моделі клінічного моніторингу. (незалежне моделювання всіх складових процесів з подальшим агрегуванням).

Огляд результатів моделювання на основі СПМ дозволив поставити нове завдання щодо удосконалення алгоритмів моделювання НЧПД при перемінних інтервалах спостережень. Удосконалення таких алгоритмів можливе за рахунок створення 2-х додаткових алгоритмічних та програмних механізмів

- M1) створення мульти моделей (мульти-нечітких і сепарабельно-реляційних моделей) та процедур процесів НЧПД,
- M2) формування схем агрегування (або поєднання) результатів застосування комбінованих алгоритмів ВКМ з результатами сепарабельно-реляційних моделей.

Розвиток і дослідження таких алгоритмів, що враховують механізми та процедури M1, M2, виконаний в цій дипломній роботі.

## 1.2. Призначення та сфера застосування результатів досліджень

Як зазначалося вище, різноманітні технологічні, виробничі, інформаційні, лікувальні та інші процеси у складних системах можуть мати значний ступень невизначеності деяких параметрів і характеристик. При цьому для них на практиці можливо отримання лише нерегулярні у часі послідовності даних (НЧПД), через різні за величинами інтервали. При тому для аналізу таких процесів є лише нерівномірні у часі послідовності даних, що показують зміни різноманітних показників систем. Важливим прикладом таких процесів є моніторинг лікування та реабілітації захворювань, функціонування багатьох складних системах тощо, які часто можуть характеризуватися високим ступенем невизначеності певних характеристик. Через указані ознаки технічних систем (ТС) на практиці можливим є лише отримання нерегулярних у часі послідовностей даних різних категорій інтервалів (детерміновані, нечіткі величини (НВ), інтервальні оцінки тощо).. Тому для аналізу доступні лише нерівномірні у часі послідовності даних, що характеризують різноманітні показники систем.

Нерівномірність інтервалів спостережень ускладнює і навіть унеможливорює моделювання та аналіз таких процесів загальновідомими методами. В дослідженнях разом з тим було встановлено, що для певних процесів зміна категорій даних та їх перетворення (агрегування, зміщення величин у часі ін.) можуть мати свої застереження, навіть бути неприйнятними, Перетворення даних може призвести до пропуску критичних станів процесів у ТС, умови виникнення критичних станів у хворих, вихід технічних систем із ладу, певні безпекові чи навіть катастрофічні події тощо. Твердження щодо необхідності проведення додаткових контрольних вимірів частопрактично не можливі, значно збільшують ресурси на моніторинг, ведуть до відмов від доходів тощо.

Іншим важливим прикладом процесів з перемінними інтервалами спостережень являються процеси моніторингу функціонування програмних

систем. Через великі труднощі та значну вартість оперативних та достовірних даних з оцінки параметрів багатьох програмних систем та програмних середовищ на практиці застосовують підходи і моделі нечіткого моделювання, аналізу і прогнозування. У якості прикладів такого типу можливо розглядати наступні завдання. Контроль показників функціонування версій деякого програмного забезпечення широкого застосування, а також оцінка ефективності програмних комплексів із розробки ігрових платформ. Для таких об'єктів встановити певні детерміновані послідовності отримання та значення контрольованих параметрів не представляється можливим. Інший приклад процесів категорії НЧПД – експлуатація та обслуговування парків ТС на основі параметрів поточного стану. В них за даними попередніх нерівномірних показників визначається раціональна черговість обслуговування груп ТС

Головними завданнями, які вирішуються за такими послідовностями НЧПД являються: прогнозування максимального (нечіткого) періоду до подій, які відповідають заданим вимогам, а також визначення певних закономірностей зв'язків між встановленими у дослідженні величинами.

У статті [30] для таких процесів НЧПД була представлена модель СПМ, яка має відмінність у окремому представленні послідовностей величин показників і інтервалів між окремими спостереженнями. В ній вперше були використані спеціальні сепарабельні форми обліку часових інтервалів між рівнями часового ряду (ЧР). В них інтервали спостережень виділяються у окремому складову моделі. Ця складова моделюється окремо, На результати моделювання таких інтервалів «накладаються» результати моделювання інших характеристик у відповідності до порядку їх розміщення у початкових НЧР. Такий комплекс параметрів визначає у моделі НЧР як значення нового моменту виникнення чергової події спостережуваного процесу, так і характеристики процесу.

Аналіз попередніх результатів моделювання процесів моніторингу реабілітаційних процесів з нерівномірними у часі подіями разом с тим показав

необхідність подальших докладніших розробок алгоритмів НЧПД. Такі висновки були зроблені тому, що періоди в моделі моніторингу клінічних процесів надзвичайно нерівномірні та особливі для різних хворих. Фактично кожний пацієнт має індивідуальну модель процесу реабілітації.

Для демонстрації структури процесів, особливостей і можливостей СПМ на рис. 1.19 та рис. 1.20 показані розраховані складові цієї моделі, а на рис. 1.21 підсумковий результат для певного пацієнта (інтервал між вимірами і рівень показника ППП, представлені в певному масштабі. При моделюванні послідовності рис. 1.19 та рис. 1.20 отримані при розмноженні первинних даних клінічного моніторингу, з використанням методу «бутстреп».

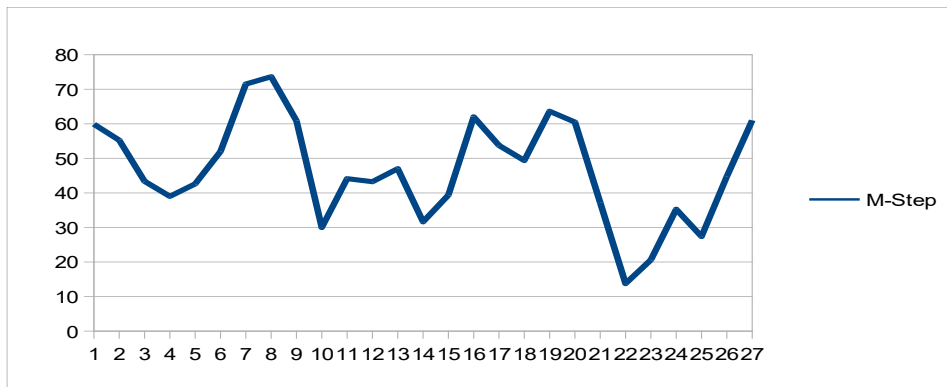


Рисунок 1.19 Графік результатів моделювання інтервалів процесу моніторингу хворого

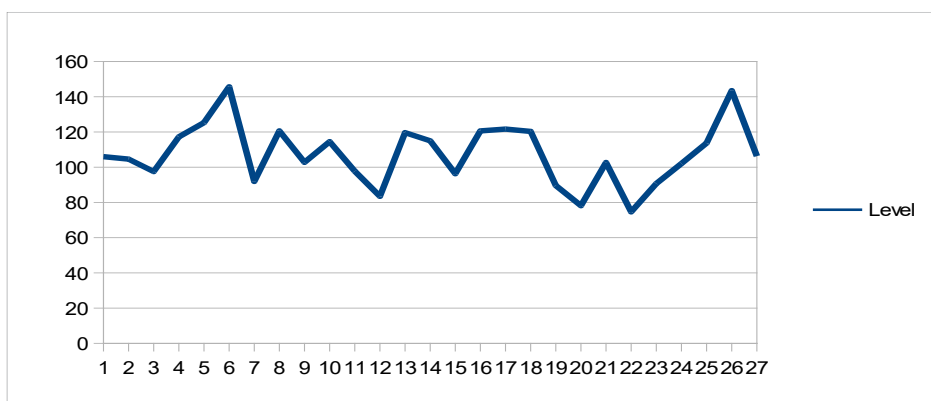


Рисунок 1.20 Графік результатів моделювання контрольованого показника моніторингу хворого (порядок вимірів)

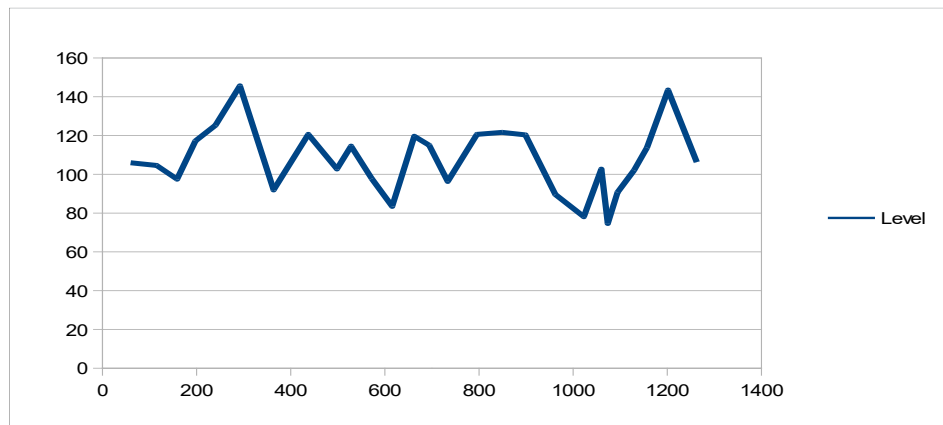


Рисунок 1.21 Графік сепарабельної модель процесу моніторингу (час)

Відзначимо особливості завдань, тому що для кожного процесу індивідуального моніторингу необхідно окремо визначати параметри моделей процесів (1.15) – (1.17). Зазначене являється необхідною вимогою, яку треба врахувати при удосконаленні версій алгоритмів та програмних засобів. Для удосконалення математичного та програмного забезпечення процесів НЧПД необхідно розробити нові форми алгоритмів і програму, які реалізують змішану форму моделювання послідовностей з нерівномірним кроком, а саме мульти алгоритми процесів НЧПД. При цьому буде застосована модель нечіткої регресії першого та другого порядку для вибору прогнозованої моделі нечіткого показника. Така форма дозволяє виконувати розрахунки як тренду процесу, так і очікуваних результатів прогнозних розрахунків. При цьому результати розрахунків за цією моделлю нечіткої регресії поєднуються з результатами моделей, які зазначені вище.

### 1.3. Постановка завдань дослідження

В розділі 1 визначено, що в багатьох завданнях при аналізі та моделюванні складних систем як вихідні дані можливо отримати лише часові послідовності характеристик процесів, часові ряди (ЧР). При тому зараз моделі та методи аналізу та моделювання ЧР з перемінним кроком спостережень є не достатньо дослідженими. Тому розвиток досліджень і розробок засобів програмного забезпечення щодо моделей, алгоритмів і програмних засобів, призначених для аналізу нерівномірних у часі послідовностей спостережень, з

метою підвищення ефективності та точності методів та алгоритмів, має науковий і практичний інтерес.

Аналіз існуючих алгоритмів, які застосовуються для процесів НЧПД дозволив визначити наступне

- 1) один із головних параметрів квантильного алгоритму на основі тренду «каппа», від значень якого суттєво залежить результат моделювання, являється евристичним, спосіб визначення значень не встановлений, також і засоби автоматизованого вибору,

- 2) для удосконалення та підвищення чисельної ефективності алгоритмів НЧПД важливим напрямком розвитку являється дослідження схем формування комбінованих процесів, які поєднують різні схеми використання параметрів первинних процесів. .

- 3) актуальним завданням розвитку є створення мульти моделей (мульти-нечітких і сепарабельно-реляційних моделей) та процедур процесів НЧПД,

- 4) важливою складовою удосконалення алгоритмів НЧПД є формування схем агрегування результатів комбінованих алгоритмів ВКМ з результатами сепарабельно-реляційних моделей.

Метою роботи являється це підвищення ефективності та точності методів і алгоритмів аналізу та прогнозування характеристик недетермінованих послідовностей спостережень з нерівномірним кроком. Для досягнення мети необхідно виконати завдання, які складаються з наступного :

- розробити алгоритмічні та програмні засоби та виконати порівняльний аналіз чисельної ефективності алгоритмів моделювання нерівномірних у часі недетермінованих числових послідовностей даних спостережень.

- розробити алгоритмічні та програмні засоби та виконати прогнозування часових послідовностей,

- розробити інструментальне програмне середовище, яке забезпечує дослідження алгоритмів моделювання нерівномірних у часі недетермінованих числових послідовностей;
- виконати порівняльний аналіз алгоритмів моделювання нерівномірних у часі послідовностей щодо точності та чисельної ефективності;
- отримати рекомендації стосовно застосування процедур моделювання недетермінованих послідовностей з нерівномірним кроком спостережень.
- розробити мульти-нечітку і реляційно-сепарабельну моделі прогнозування послідовностей з перемінним кроком,
- удосконалити процедуру формування комбінованих алгоритмів моделювання досліджуваних процесів,
- розробити програмні засоби для моделювання та прогнозування характеристик числових послідовностей з перемінним кроком спостережень,.

## Висновки до розділу 1

Аналіз технологічних, інформаційних, економічних процесів, процесів моніторингу та інших процесів, представлених ЧР, показав значну актуальність реалізації завдань моделювання ЧР з перемінним кроком спостережень, тобто НЧПД. Через це визначена мета роботи, яка полягає у підвищенні ефективності та точності методів і алгоритмів аналізу та прогнозування характеристик НЧПД послідовностей спостережень з нерівномірним кроком. В розділі проведено аналіз предметної сфери моделювання недетермінованих процесів, зроблено опис особливостей завдань моделювання, визначена актуальність дослідження, а також зроблено огляд результатів основних наукових досліджень і засобів програмних аналогів. Також наведено призначення та сфера застосування результатів досліджень. На підставі цих результатів зроблена постановка завдань дослідження. При цьому також сформульовані завдання щодо дослідження та необхідних розробок моделей та алгоритмів, засобів програмного забезпечення.

## 2. ОБГРУНТУВАННЯ ТА МЕТОДИКА ДОСЛІДЖЕНЬ АЛГОРИТМІВ ІЗ МОДЕЛЮВАННЯ ТА ПРОГНОЗУВАННЯ НЕДЕТЕРМІНОВАНИХ ПРОЦЕСІВ З ПЕРЕМІННИМ КРОКОМ

2.1. Дослідження ефективності алгоритмів із моделювання та прогнозування числових послідовностей з перемінним кроком.

Однією з важливих проблем використання математико-статистичних і економетричних методів прогнозування є обмеження на обсяг вибірки. На жаль не завжди можливо представити вибірку даних потрібних розмірів. По-перше, велике нагромадження статистичних даних здійснюється поступово і, внаслідок цього, обсяги вибірок ще малі. Для роботи з малими обсягами вибірок потрібно мати відповідні методи обробки даних. По-друге, достатніх обсягів вибірок може і не бути (наприклад, в експериментальних, індивідуальних, дрібносерійних, серійному виробництвах). Наприклад, обсяг спостережень за попитом на товар невеликий, тому що товар лише нещодавно з'явився на ринку, або досліджуємо показники підприємства за два роки, використовуючи квартальні дані. Втім слід не тільки отримати кількісні прогнозні характеристики за допомогою моделей, але й вивчити статистичні властивості вибірок або оцінок параметрів моделей. Вирішити ці проблеми можна за допомогою методу «бутстреп», який запропонований у 1977 р. Б. Ефроном за Стенфордського університету (США).

Б. Ефрон розробив новий спосіб розмноження вибірок, що істотно використовую датчики псевдовипадкових чисел. А саме, він запропонував будувати нові вибірки, моделюючи вибірки з емпіричного розподілу. Іншими словами, узяти кінцеву сукупність з  $n$  елементів вихідної вибірки  $x_1, x_2, x_3, \dots, x_{k-1}, x_k, x_{k+1}, \dots, x_{n-1}, x_n$  і за допомогою датчика випадкових чисел сформувати з неї будь-яке число розморожених вибірок.

2.1.1. Процедури аналізу та прогнозування характеристик на основі квантильних алгоритмів.

В розд. 1.3 були зазначені певні недоліки існуючих квантильних алгоритмів (1.2) – (1.7) та (1.9) – (1.12), які застосовуються для процесів НЧПД, при цьому було встановлено, що один із головних параметрів квантильного алгоритму «каппа», являється евристичним, не визначається спосіб отримання його значень. Разом з тим від значень «каппа» суттєво залежать загальні результати моделювання та аналізу НЧПД, можливості удосконалити алгоритми та засоби щодо його автоматизованого розрахунку. Наступним та іншим шляхом щодо удосконалення квантильних алгоритмів (1.2) – (1.7) та (1.9) – (1.12), а також підвищення їх числової ефективності являється формування та дослідження нових схем комбінування результатів реалізації квантильних алгоритмів, тобто поєднують різні алгоритми використання параметрів первинних процесів. .

На рис. 2.2 приведена загальна структура класичного квантильного алгоритму (1.2) – (1.7), яка відображає процедуру визначення тренду процесу, який досліджується, а також розрахунку прогнозних значень на наступному кроці, новий рівень процесу, що враховує показник «тренд».

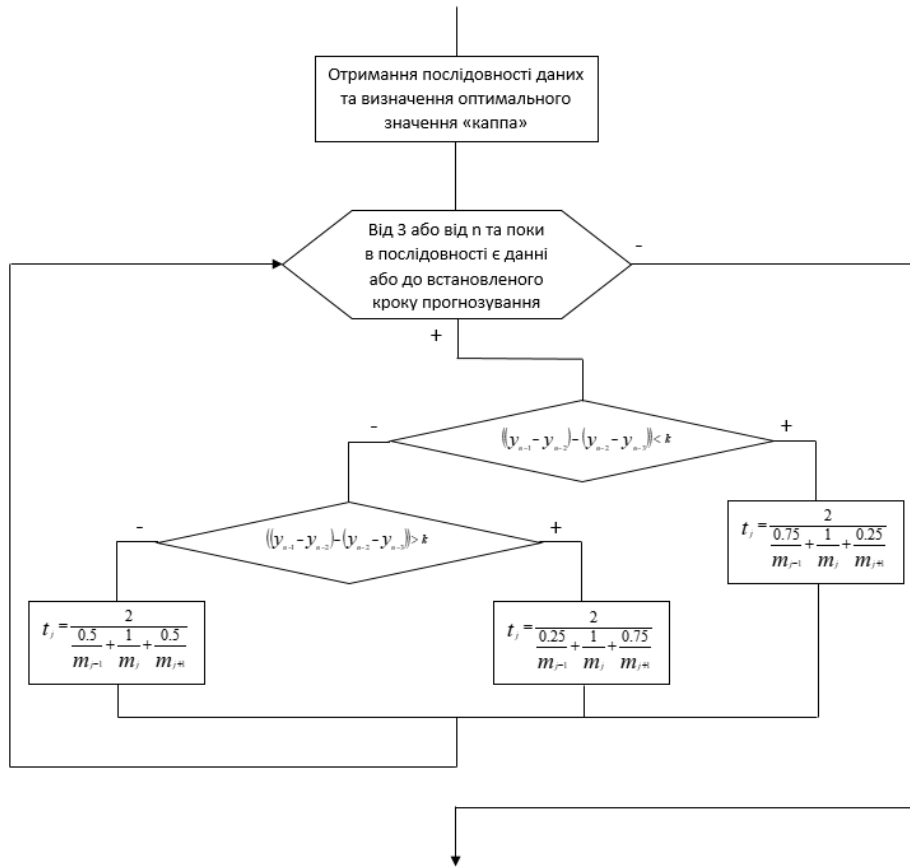


Рисунок 2.2 Узагальнена структура базового квантильного алгоритму

Відповідно до (1.2) – (1.4) при визначенні показника «тренд» необхідно мати значення «каппа». В роботі були проведені числові дослідження, з використанням даних процесів TAIFEX-T та TAIFEX, в яких досліджувалися оцінки середньої похибка для відомих квантильних

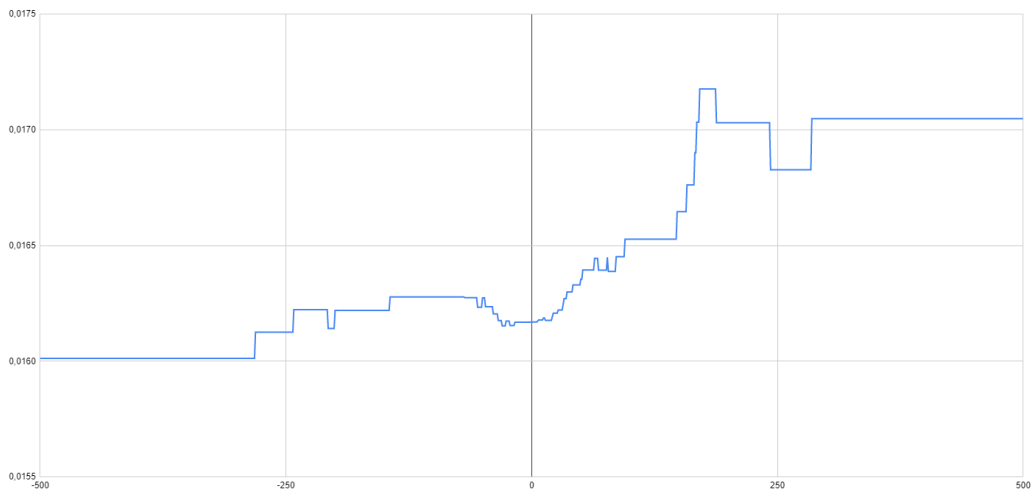


Рисунок 2.3 Середня похибка для квантильного алгоритму

алгоритмів, рис. 2.3, а також для модифікованого квантильного алгоритму, який передбачав композицію окремих форм аналізу процесу, рис. 2.4..

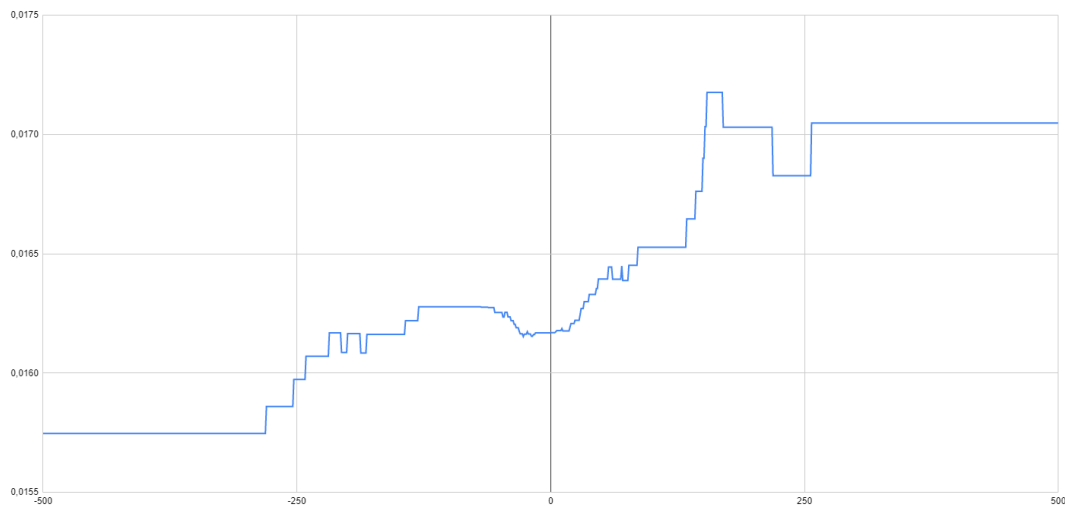


Рисунок 2.4 Середня похибка для удосконаленого алгоритму

Графіки показують суттєві залежності середньої точності (похибки відтворення вихідної послідовності) від параметру «каппа», який відкладений пр осі  $OX$ . При цьому загальна тенденція задежності суттєво не змінилася. Таким чином можна зробити висновок, що алгоритми комбінування, які розроблені та досліджені в дипломній роботі, мають таку саму залежність від «каппа», як і первинна квантильна модель. Що підтверджує її актуальність та можливість застосування.

Разом с цим результати аналізу залежності середньої похибки від константи оцінювання парасметру «тренд» дозволяють достатнім вважати залежність такою, що може бути замінена лінійною моделлю. Значить при автоматичному виборі значення «каппа» можливо застосовувати алгоритми одновимірної оптимізації, призначені для розрахунку цієї константи. За рахунок отриманих результатів стає можливим формування програмної процедури налаштування квантильних алгшоритмів, використовуючи дані історичних послідовності, або ж розмножені дані за методом «бутстреп». Блок-схема цієї процедури наведена в розділі 2.1, дивись рис. 2.1.

## 2.1.2. Аналіз напрямків удосконалення алгоритмів на основі нечіткого моделювання процесів, комбіновані алгоритми обчислювальних процесів

В якості інших способів модифікації та удосконалення квантильних алгоритмів (1.2) – (1.7) та (1.9) – (1.12) є формування нових схем комбінування розрахунків, виконаних за квантильними алгоритмами, що позначені у розд. 1.1.2, моделі (1.15) – (1.17). В таких комбінованих моделях передбачається, що значення параметру змішування « $\alpha$ » підбирається на основі процедури мінімізації квадрату середньої похибки. Результати попереднього аналізу на основі розрахунків для таких спеціалізованих схем алгоритмів моделювання показали суттєвість завдання із визначення параметру змішування « $\alpha$ ». Визначення значення його необхідно виконувати для кожного окремого часового ряду, в чому полягає процедура налаштування удосконаленого комбінованого алгоритму. Розрахунок оптимального значення параметру « $\alpha$ » змістовно являється налаштуванням процедури аналізу та прогнозування. При цьому виконується процедура оптимізації на основі одновимірного пошукау. Блок-схема алгоритму з визначення оптимального значення параметру « $\alpha$ » приведена на рис. 2.5.

### **Алгоритм налаштування параметрів моделі прогнозування (1.15) – (1.17)**

$$W(w) = wV(Y) + (1 - w)V(\Delta) \quad (2.1)$$

$$0 \leq w \leq 1 \quad (2.2)$$

Результат  $w_x$

Дані:  $Y(k)$ ,  $Y_1(Y(k))$ ,  $Y_2(\Delta(k))$ . Алгоритм  $Y_m(k)$  визначається таким чином

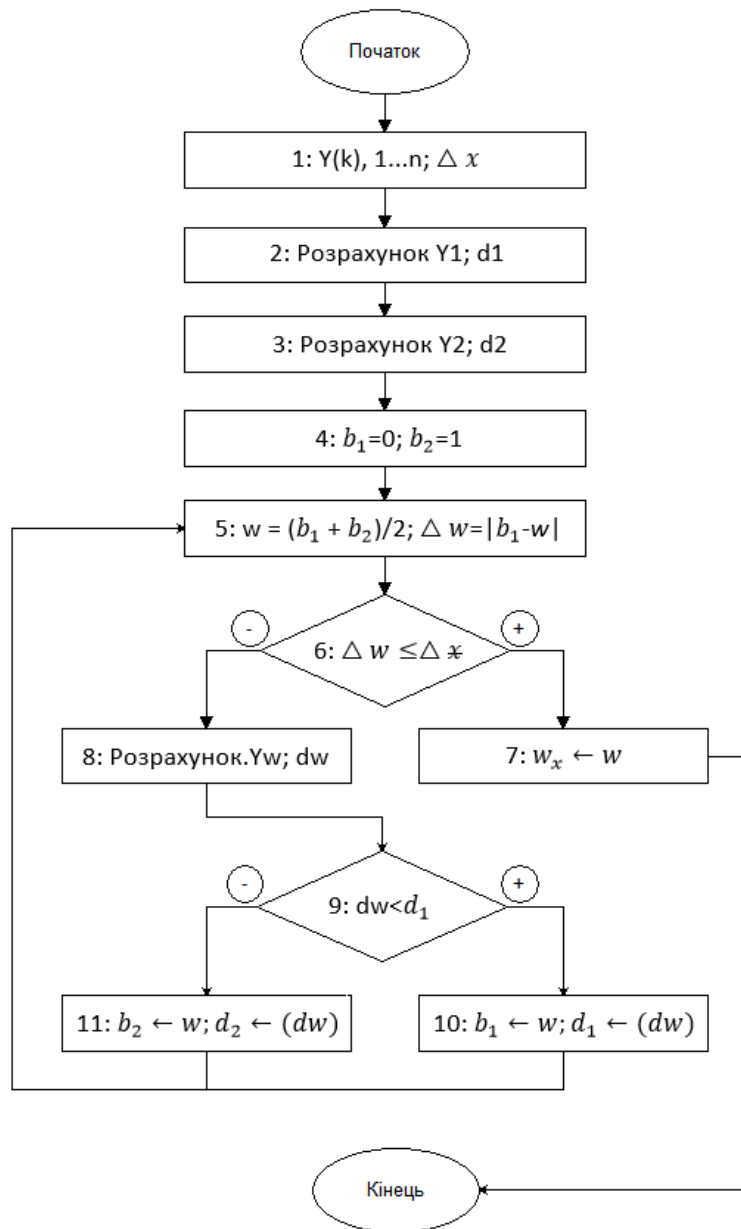


Рисунок 2.5 Блок-схема процедури з визначення оптимального значення параметру « $w$ » при налаштуванні параметрів комбінованих квантильних алгоритмів моделювання НЧПД

На рис. 2.5 позначено наступне: Результат оптимальна величина  $w_x$

Дані:  $Y(k)$ ,  $Y1(Y(k))$ ,  $Y2(\Delta(k))$ . Алгоритм  $Y_m(k)$

Алгоритм налаштування параметрів моделі прогнозування як процедура одновимірної оптимізації, коли

$$W(w) = wV(Y) + (1 - w)V(\Delta), 0 \leq w \leq 1.$$

2.1.3. Реляційно-сепарабельна модель моделювання і процедури дослідження чисельної ефективності та точності алгоритмів.

В дипломній роботі отримано розвиток алгоритмів аналізу та прогнозування НЧПД з перемінним кроком спостережень, за рахунок формування нової реляційно-сепарабельна моделі (PCM), а також мульти-нечіткої квантильної моделі процесів з нерівномірним кроком між вимірами, рівнями. Головний зміст такого підходу полягає в тому, щоб використовувати не лише сепарабельну модель СПМ, а комбінувати її результати з PCM. На відміну від СПМ, в яких кожний параметр формується незалежно від інших, а вони агрегуються на основі номерів спостережень, див. модель СПМ (1.1), в PCM формується нечітка модель зв'язків між кроками у часових інтервалах та змінами контрольованого параметру процесів НЧПД. При цьому як параметри перемінних інтервалів, так і величини контрольованих параметрів замінюються системами дискретних нечітких величин. Модель PCM відтворює нечітку регресію, яка дозволяє на основі поточних та попередніх значень (номери термів нечіткої величини) характеристик процесу визначити, до яких нечітких величин процес «перейде» на наступному своєму кроці, етапі. То ж за попередніми НВ будуть визначені терми, які визначають очікувані на наступному кроці процесу НВ, що дозволяє виконувати аналіз та прогнозування НЧПД.

При формуванні PCM будемо використовувати такі позначення:

- нечітка реляційна модель  $R(T, V)$  процесів  $\{T, V\}$ , тобто  $V = A(T)$ ,
- параметри процесів  $\{T, V\}$ :  $T$  – це процес відтворення інтервалів між спостереженнями, процес  $V$  – процес зміни рівнів контрольованого параметру (або сукупності параметрів), який відбувається у відповідності до етапів процесу « $T$ ».

Алгоритм формування нечіткої реляційної моделі  $R(T, V)$  процесів  $\{T, V\}$  формується наступним чином. .

Дані: Нечіткі моделі процесу  $\tilde{T}$  і  $\tilde{V}$ :

Модель  $\tilde{T}$ : характеризується наступним чином. Задані

– вектор номерів змінних апроксимацій  $TN(k)$

– вектори моделі трикутної величини на кроці «k»

$$T \Delta (k)(a_k, b_k, c_k), \{Ta(k), Tb(k), Tc(k)\}$$

– вектор значень приналежності величини  $T \Delta (k) \rightarrow TM(k)$

$k = 1, 2, \dots, n$ . Вектор встановлює апроксимації нечіткими трикутними величинами часових інтервалів між спостереженнями.

Кроки процесу II.

Модель  $\tilde{V}$ : характеризується подібним же чином. Задані

– вектор номерів змінних апроксимацій  $VN(k)$

– вектори моделі трикутної величини

$$V \Delta (k)(a_k, b_k, c_k)\{Va(k), Vb(k), Vc(k)\}$$

– вектор значень приналежності величини  $V \Delta (k) \rightarrow VM(k)$

$k = 1, 2, \dots, n$ .

Кроки процесу III.

Розмірності нечітких моделей апроксимації  $(m_t, m_v)$

Етап IV. Задаються початкові значення процесів НЧПД

$(T_0; \Delta T)$  – початкове значення  $T$ ,

$(V_0; \Delta V)$  – початкове значення  $V$ .

За процедурою для кожного кроку послідовностей  $TN(k)$  та  $VN(k)$  виконується визначення  $V \Delta (k) \rightarrow VM(k)$  та  $T \Delta (k) \rightarrow TM(k)$ . На основі цих даних створюється нечітка реляційна (таблична) модель. При цьому в результаті формуються дві таблиці для кожного параметра процесу  $\{T, V\}$ .

Результат: Формування таблиць реляційних зв'язків  $R(T, V)$ .

$RT(T, V)$ , де:

$r_{ij}$  – номер  $\langle q \rangle$  наступної НВ для  $T$

$i$  – номер поточного  $k$  значення  $\Delta T$

$j$  – номер поточного  $k$  значення  $\Delta V$

$RM(T,V)$  – значення  $M_{ij}$  нового кроку.

Таблиця  $RT$  дає номер нечіткої змінної.

Таблиця  $RM$  – дає ступінь належності нового значення.

Схема алгоритму формування приведена на рис. 2.6. Вона передбачає виконання наступних функцій, за номерами блоків.

1. Обнулити  $RT, RMT, RV, RMV$

2. Цикл по  $\langle k \rangle, k = 1, \dots, n-1$

Формування індексів для  $RT, RM$

$i = TN(k)$  {номер змінної  $T$ }

$j = VN(k)$  {номер змінної  $V$ }

$i_1 = TN(k+1); M1 = TM(k+1)$

Формування таблиць  $RT, RM$ :

if  $M1 > RM(i,j)$  then

{

$RMT(i,j) \leftarrow M1;$

$RT(i,j) \leftarrow i_1;$

}

3. Подібним чином сформувати  $RV, RMV$ .

4. Виклик програми  $ASt$  для розрахунків прогнозних кроків послідовності  $T \Delta (n + p), p = 1, 2, \dots, m_p$

5. Прогнозування змінних  $\Delta T, \Delta V$  за таблицею  $RT, RV/$

Продовжувати послідовності  $TN(k), VN(k), k = n+1, n+2, \dots, n+p$

За  $\langle k \rangle$  визначаємо  $i = TN(k); j = VN(k);$

Далі  $TN(k+1) = RT(i,j)$

$TV(k+1) = RV(i,j)$

і далі ... до значення  $\langle k \rangle = p$ .

На рис. 2.6 приведена узагальнена схема формування реляційного відношення, представленою системою таблиць  $RT$ ,  $RMT$ ,  $RV$ ,  $RMV$ . Зіст блоків стисло позначено в схемі, і складається з такого

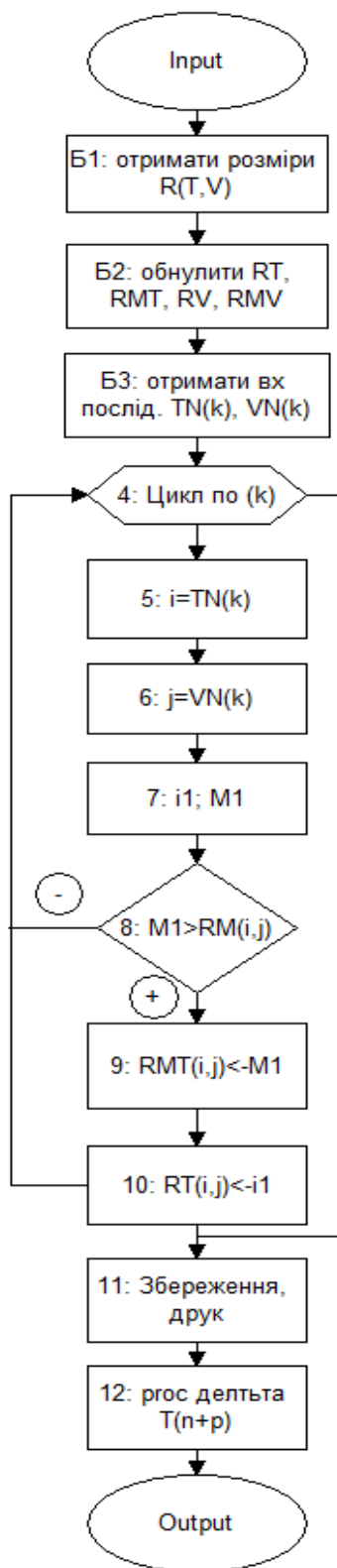


Рисунок 2.6 Схема формування реляційного відношення  $RT$ ,  $RMT$ ,  $RV$ ,  $RMV$

На початку виконання алгоритму необхідно отримати розміри відношення  $R(T,V)$  та встановити початкові значення в таблиці  $RT, RMT, RV, RMV$ . Після отримання вхідних даних про первинні послідовності  $TN(k), VN(k)$  організується цикл за їх елементами  $(k)$ . В ньому виконується формування індексів для  $TN(k)$  та для  $VN(k)$ . При формуванні таблиць  $RT, RM$  перевіряється умова  $M1 > RM(i,j)$ , що визначає вимогу збереження найбільшого значення ступні приналежності. Отримані дані стосовно рогнгозних (наступних) величин моделі НВ діапазонів записуються в таблиці – внесення даних  $M1$  в таблицю  $RMT(i,j)$ , внесення даних  $i1$  в таблицю  $RT(i,j)$ . Отримані результати формування при необхідності беруться, друкуються. Процес продовжується для всіх передбачених кроків, враховуючи і кроки прогнозування.  $T(n+p)$ .

В роботі розроблено модель та процедуру аналізу та прогнозування НЧПД на основі реляційного відношення другого порядку. При цьому відмінність таких моделей і алгоритмів полягає в тому, на відміну від РСМ першого порядку, що наступний елемент (нечітка величина очікувана на наступному кроці) визначається на основі двох попередніх як для параметру часу ( $T$ ), так і для контрольованого параметру ( $V$ ).

Модель реляційного відношення другого порядку (НРМ2), представлена в табличному вигляді, табл. 2.1, формується з використанням модифікованого алгоритму формування реляційного відношення рис. 2.6 таким чином. При формуванні моделі другого порядку табл.1 процедура рис. 2.6 використовується двічі. Спочатку формуються таблиці безпосередньо для відношень першого порядку. Для моделі другого порядку використовують поточні значення вхідних (номер рядку, наприклад,  $i1$ ) та вихідних (номер стовпця, наприклад  $j1$ ) показників реляційних таблиць. При цьому на підставі значень  $(i1, j1)$ , використовуючі таблиці для показників  $T$ , визначається номер рядку прогнозу, наприклад,  $i2$ , а за таблицею вихідних

характеристик (номер стовпця, наприклад  $j_2$ ) . Після цього на підставі значень  $(i_1, j_1)$  визначаються номери величин, що визначають прогноз відповідно до моделі другого порядку, наприклад  $(i_3, j_3)$ . Саме ці значення далі зберігаються в табл. 2.1 для моделі відношення 2-го порядку

$$(i_1, i_2, j_1, j_2) \rightarrow (i_3, j_3).$$

Нечітка реляційна модель НРМ2 забезпечує відтворення таких зв'язків для нечітких величин

$$R_v(T \times T, V \times V, V); \quad (2.3)$$

$$R_t(T \times T, V \times V, T). \quad (2.4)$$

У відношенні (2.3) для послідовності значень  $T \times T$ , а також відповідних за номером  $\langle k \rangle$  рівня послідовності значень  $V \times V$  визначається (за рахунок перевірки всіх значень вихідних даних для  $\{T, V\}$ ) нечітка величина, як модель наступного кроку за контрольованим показником процесу. Для відношення (2.4) при таких самих вхідних даних визначається нечітка величина, як модель наступного кроку за часовим, інтервальним, показником процесу. Також для реалізації реляційних моделей НРМ2 необхідно мати такі базові математичні моделі апроксимації областей значень параметрів процесів. А саме для предстваленні найпростіших апроксимацій областей варіювання змінних  $\{T, V\}$  трьома термами (ліве, центр, праве  $(\widehat{S}_1, \widehat{S}_2, \widehat{S}_3)$ ) необхідно задати такі лінгвістичні змінні - НВ; FT(3); FV(3):  $(\widehat{S}_1, \widehat{S}_2, \widehat{S}_3)$ .

Структура правил нечіткої НРМ2 (при трьох термах бази апроксимації) приведена у наступних продукціях :

$$R_v: \tilde{T}_{i_1}, \tilde{T}_{i_2}, \tilde{V}_{j_1}, \tilde{V}_{j_2} \rightarrow V_{iq} (q \in \{1, 2, 3\}) \quad (2.5)$$

$$R_t: \tilde{T}_{i_1}, \tilde{T}_{i_2}, \tilde{V}_{j_1}, \tilde{V}_{j_2} \rightarrow T_{ip} (p \in \{1, 2, 3\}) \quad (2.6)$$

Модель із визначення наступної величини, яка відповідає  $R_v$  (2.3), визначається за (2.5), а для  $R_t$  (2.4) моделюються за (2.6).

Таблиця 2.1

Структура моделі реляційного відношення другого порядку

step	(n-1)	$V_1$	$V_1$	$V_1$	$V_2$	$V_2$	$V_2$	$V_3$	$V_3$	$V_3$	...
(n-1)	(n)	$V_1$	$V_2$	$V_3$	$V_1$	$V_2$	$V_3$	$V_1$	$V_2$	$V_3$	...
$T_1$	$T_1$	$\Gamma_{11,11}$									
$T_1$	$T_2$			$\Gamma_{13,12}$							
$T_1$	$T_3$							$\Gamma_{i1i2,j1j2}$			
$T_2$	$T_1$			$\Gamma_{21,13}$							
$T_2$	$T_2$										
$T_2$	$T_3$			$\Gamma_{i1i2,j1j2}$			$\Gamma_{23,23}$				
$T_3$	$T_1$										
$T_3$	$T_2$							$\Gamma_{32,31}$			
$T_3$	$T_3$										

$$r_{i1i2,j1j2} = \begin{cases} T_i & \text{для } R_t(\mathcal{X}) \\ V_j & \text{для } R_v(\mathcal{X}) \end{cases}$$

В таблиці 2.1 закодована структура правил нечіткої НРМ2,

$$R_v: \tilde{T}_{i1}, \tilde{T}_{i2}, \tilde{V}_{j1}, \tilde{V}_{j2} \rightarrow V_{iq} (q \in \{1,2,3\}), R_t: \tilde{T}_{i1}, \tilde{T}_{i2}, \tilde{V}_{j1}, \tilde{V}_{j2} \rightarrow T_{ip} (p \in \{1,2,3\})$$

а саме для наведеного прикладу 3-х складових баз представлення НВ, які представляють у нашій моделі області зміни  $\{T, V\}$ . В таблиці величини  $r_{i1i2,j1j2}$  визначаються на основі відповідних вхідних значень для параметру  $\{T\}$ , два перші стовпці, а також вхідних значень для параметру  $\{V\}$ , два перші рядки табл. 2.1. В залежності від моделей продукцій (2.5) – (2.6) змінні визначають величини  $T_i$  для  $R_t$  або  $V_j$  для  $R_v$ .

Процедура прогнозування на основі моделей процесу (2.3) – (2.6) при розрахованні реляційній моделі виду табл. 2.1 відбувається таким чином. Нехай для поточного кроку процесу  $\{T, V\}$  відомі попередньо визначені значення  $\tilde{T}_{i1}, \tilde{T}_{i2}, \tilde{V}_{j1}, \tilde{V}_{j2}$ . Тоді на основі продукційних моделей (2.5)-(2.6) можуть бути

визначені НВ для  $V_{iq}, T_{ip}$ . За допомогою базових моделей цих НВ можна розрахувати (функція дефазифікації) числові параметри наступного кроку процесу. Це в свою чергу забезпечить можливість перейти до нового  $\langle k+1 \rangle$  кроку (та до оцінки значень відповідних рівнів) процедур аналізу або прогнозування.

Для реалізації зазначеної форми алгоритмів НРМ2 необхідно мати відповідну структуру програмної реалізації, яка забезпечує утворення та розрахунки за наведеними схемами.

2.1.4. Мульти-нечітка квантильна модель недетермінованих послідовностей з перемінним кроком.

Базова квантильна модель (1.2) – (1.13) процедури FTS третього порядку з трендом рис.2.2 використовує певну форму нечіткого моделювання, апроксимацію, області досліджуваного недетермінованого процесу. При тому вибір саме структури нечіткого представлення процесу залишається поза наступними алгоритмами аналізу та прогнозування. Наприклад, модель (1.2) – (1.13) може використовувати наступну апроксимації області варіювання показника  $\Delta$

Область значень  $\Delta V$  Моделювання лінгвистичної перемінної  $\tilde{\Delta V}$

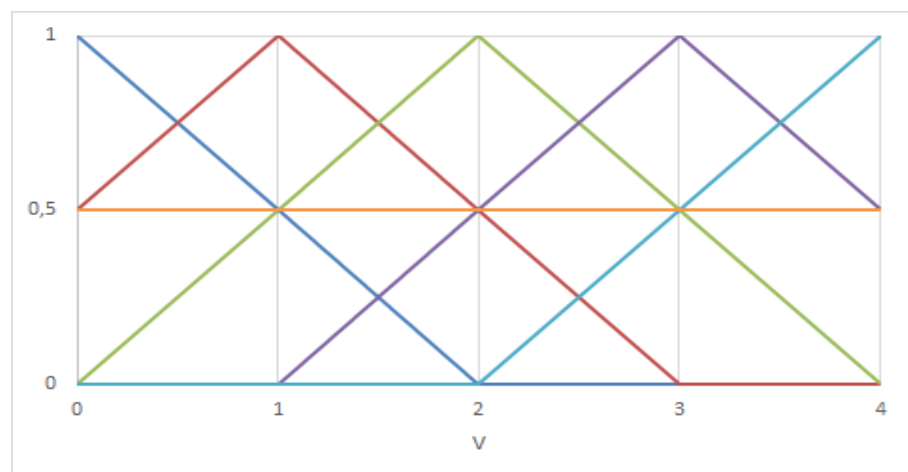


Рис. 2.7 Нечіткі моделі термів лінгвистичної перемінної області варіювання  $\tilde{\Delta V}$

Всі нечіткі моделі термів, складових «Діапазон  $\Delta$ », являються трикутними нечіткими величинами, а саме відповідно рис. 2.7 позначаються  $S_1=\Delta_1$ ,  $S_2=\Delta_2$ .  $S_3=\Delta_3$ .  $S_4=\Delta_4$ .  $S_5=\Delta_5$ .

Моделі квантильного алгоритму (1.4) – (1.7) використовують формули квантильної регресії саме для трикутних моделей нечітких множин рис. 2.7. При тому результат моделювання процесів та прогнозування наступних рівнів виконується через значення центрів трикутних квантилів. Зрозуміло, що при інших структурах апроксимації області варіювання результати розрахунків будуть відмінними, хоча процедура рис. 2 не зміниться. Рівняння (1.4) – (1.7) використовують центри нечітких квантилів для розрахунку параметрів апроксимованої моделі процесу. Виходячи з цих міркувань можливо формування кількох апроксимацій досліджуваного процесу. Які будуть представлені через різні центри квантилів. Наприклад, в якості найпростішої можна вибрати апроксимацію за трьома трикутними квантилями -  $S_1=\Delta_1$ ,  $S_3=\Delta_3$ .  $S_5=\Delta_5$ . Таким чином буде утворена мульти-нечітка модель процесу, в якій в кожній точці процесу вираховуються та змішуються значення характеристик через різні набори центрів квантилів. Далі отримані значання узагальнюються за допомогою вагових коефіцієнтів. Наприклад, при рівних значеннях значень для кожного процесу, тобто як центр ваги. При тому весь алгоритм мульти-нечіткої моделі залишається таким самим.

В нашій роботі була сформована і досліджена мульти-нечітка модель квантильного алгоритму, яка складалася з трьох компонентів

M1 – трикутна модель з квантилями

$$S_1=\Delta_1, S_2=\Delta_2. S_3=\Delta_3. S_4=\Delta_4. S_5=\Delta_5.$$

M2 - трикутна модель з квантилями

(2.7)

$$S_1=\Delta_1, S_3=\Delta_3. S_5=\Delta_5.$$

M3 – квантильна модель зі значеннями

показників процесу (1.12) – (1.14).

В якості вагових коефіцієнтів щодо узагальнення результатів моделей M1, M2, M3 досліджувалися дві форми – при рівних вагових коефіцієнтах, при вагових коефіцієнтах, які розраховуються на основі відстані значення розрахованого за моделлю показника від центру квантиля. Така друга форма обрана через те, що базові квантильні моделі спрямовані на відображення саме центрів своїх нечітких трикутних величин.

Іншим способом налаштування параметрів узагальнення мульти-нечіткої моделі (2.7) та подальшого узагальнення комбінованих квантильних алгоритмів на основі «тренду» (1.2) – (1.7) і (1.9) – (1.12) являється визначення коефіцієнтів узагальнення для (2.7) на основі процедури мінімізації квадрату середньої похибки. При цьому вирішується завдання оптимізації математичної моделі структури виду (2.1) – (2.2), яке на відміну від нього використовує результати моделей M1, M2, M3 (2.7). Подібний алгоритм налаштування параметрів застосовувався для моделі прогнозування (1.15) – (1.17). Розрахунок оптимальних значень параметрів моделі узагальнення змістовно являється налаштуванням мульти-нечіткої процедури аналізу та прогнозування. Для цього використовуються дані моделей M1, M2, M3 (2.7) для попередніх рівнів досліджуваного процесу, відповідно центрів яких розраховуються параметри відхилень. Для мульти-нечіткої моделі, наприклад, з моделями M1 та M3 для розрахунку коефіцієнтів узагальнення можна використовувати процедуру, приведену на блок-схемі рис. 2.5.

2.2. Дослідження функціональної ефективності алгоритмів моделювання недетермінованих послідовностей з перемінним кроком.

Метою дослідження функціональної ефективності алгоритмів моделювання недетермінованих послідовностей з перемінним кроком було встановлення властивостей достовірності та числової ефективності запропонованих в роботі модифікації квантильних алгоритмів, а також алгоритмів налагодження параметрів квантильних алгоритмів («каппа» та « $\alpha$ »)

та алгоритмів комбінування (1.15) – (1.17). Для проведення таких досліджень були виконані реалізації процесів TAIFEX в середовищі Excel. На основі цих даних досліджень було підтверджено, що запропоновані мульти-нечіткі моделі (2.7) та модифікації комбінування апроксимацій процесів пошуку дають достовірні числові результати, при тому забезпечують в певних випадках підвищення точності моделювання (зменшення середньої похибки). Також було встановлено, що застосування моделі реляційного відношення другого порядку дозволяє виконувати завдання моделювання, аналізу та прогнозування процесів з перемінним інтервалом типу НЧПД. Слід зазначити також підвищення ефективності алгоритмів СПМ в разі, коли виконувалися комбіновані розрахунки за СПМ разом з результатами алгоритмів НРМ2.

Результати проведених досліджень дали можливість показати ефект і оцінювання ефективності зазначених вище алгоритмів (2.7), а також комбінування та узагальнення результатів різних моделей. Такий підсумок дав підстави для виконання їх програмної реалізації.

2.3. Особливості структури процесів моделювання числових послідовностей з перемінним кроком.

Особливості структури процесів моделювання числових послідовностей з перемінним кроком визначені тим, що при цьому застосовується сепарабельна модель, що наведена вище, розд 1. В ній незалежно між собою виконується моделювання рівнів показників контролю (можливо використовувати кілька характеристик процесів) і інтервалів між спостереженнями. Суттєвою особливістю процесів клінічного моніторингу, які вибрані в якості предмкту досліджень, являється їх мала довжина (кількість рівнів спостережень), а також «індивідуальність», коли для кожного хворого необхідно мативласну і відмінну від загального модел. З метою забезпечення можливості моделювання таких коротких послідовностей (клінічних моніторинг конкретного хворого) необхідно спочатку дані розмножувати, в

нашому випадку методом «бутстреп». Як раніше було зазначено, цей метод дозволяє отримати вибірки з більшим числом членів, а також із збереженням тих же властивостей процесу., що містить коротка послідовність даних.

З метою урахування комплексних властивостей первинних даних процесів НЧПД моніторингу, а також їх явного представлення в моделі, крім аксіом сеперабельної моделі, в цій роботі запропоновані нова мульти-нечітка квантильна модель (2.7), а також нечітко-реляційна модель (НРМ), призначені для моделювання такого типу НЧПД процесів. В роботі досліджено можливості моделей 2-го (другого) порядку. При тому порядок НРМ залежить від кількості попередніх етапів, які ураховуються при моделюванні та прогнозуванні. Наприклад, (1.5) – (1.7) і (1.12) – (1.14) являються моделі

Засобами створюваного програмного комплексу передбачається виконати дослідження структури моделей нечітких величин, числа квантилів, нечітких величин, на які необхідно розділити області варіювання показників, при яких буде забезпечуватися достатня точність аналізу та прогнозування параметрів процесів, що спостерігаються.

## Опис структури засобів програмно-апаратного середовища

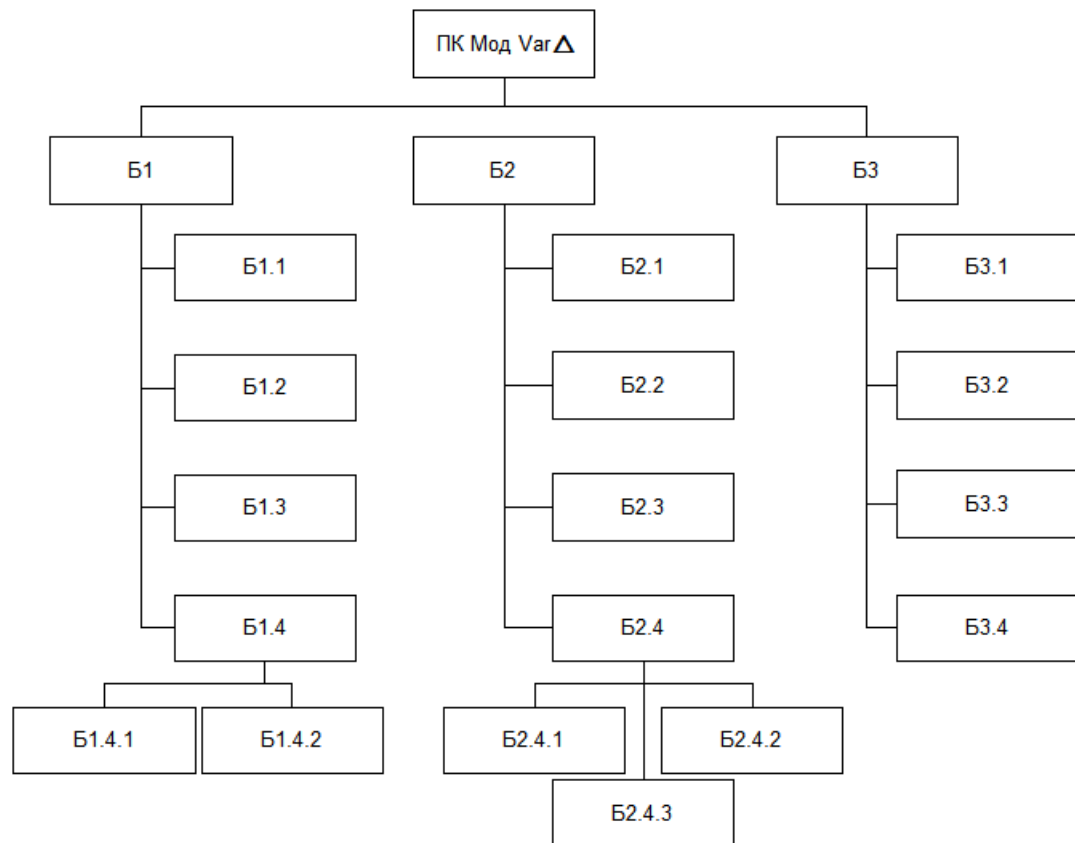


Рисунок 2.7 – Структура програмного комплексу аналізу та моделювання процесів з нерівномірним кроком спостережень.

Для реалізації методів моделювання та прогнозування з перемінним кроком між спостереженнями в рамках сепарабельної моделі (СПМ) було проведено розробку програмного комплексу «Дослідження методів моделювання недетермінованих числових послідовностей з перемінним кроком спостережень».

На рисунку рис. 2.7 наведена загальна структура програмного комплексу моделювання процесів, представлених спостереженнями з нерівномірним кроком між спостереженнями. При цьому блоки визначають наступне:

Б1 – Комплекс функцій щодо забезпечення управління даними.

Б1.1 – Отримання даних досліджуваного процесу, представленого послідовністю дійсних чисел (Н), де – послідовність «Н» визначається

величинами показника, упорядкованими за номерами спостережень ( $h_1, h_2, \dots, h_k, \dots, h_n$ ).

Б1.2 – збереження, контроль, формування представлення даних (Н).

Б1.3 – функції щодо відображення даних процесів (Н).

Б2 – комплекс функцій із моделювання і прогнозування значень показників (Н, Т), Н(Т).

Б2.1 – аналіз та прогнозування (Н), за алгоритмом квантильної моделі процесу.

Б2.2 – аналіз та прогнозування (Н), за алгоритмом модифікованої квантильної моделі процесу.

Б2.3 – аналіз та прогнозування (Н), за алгоритмом комбінування моделей процесу, визначення параметру логістичного відображення « $\alpha$ ».

Б2.4 - аналіз та прогнозування (Н), за алгоритмом реляційно-сепарабельного моделювання (PCM) процесу.

Б2.4.1 – налаштування параметрів алгоритмів PCM.

Б2.4.2 – аналіз процесів (Н, Т) алгоритмами PCM різного рангу (rank 0, rank 1, rank 2).

Б2.4.3 – прогнозування процесів (Н, Т) алгоритмами PCM різного рангу (rank 0, rank 1, rank 2) на встановлений період.

Б3 - комплекс функцій із відображення результатів моделювання і прогнозування значень показників (Н, Т), Н(Т).

Б3.1 – відображення даних для моделювання (Н, Т), Н(Т).

Б3.2 – відображення даних результатів сепарабельного моделювання (Н, Т), Н(Т).

Б3.3 – відображення даних результатів реляційного моделювання процесів (Н, Т), Н(Т).

Б3.4 – відображення даних результатів реляційно-сепарабельного моделювання процесів (Н, Т), Н(Т).

Блок-схема рис. 2.7 урахує всі перелічені вище особливості структури процесів моделювання числових послідовностей з перемінним кроком, які досліджені в дипломній роботі.

## Висновок до розділу 2

У розділі отримано розвиток та проведено дослідження можливостей та ефективності алгоритмів із моделювання та прогнозування часових послідовностей НЧПД з нерівномірним кроком. Результати розділу містять наступне. Проведені дослідження ефективності алгоритмів із моделювання та прогнозування числових послідовностей з перемінним кроком. Розроблений алгоритм процедури розмноження даних методом «бутстреп», досліджені процедури аналізу та прогнозування характеристик процесів на основі квантильних алгоритмів. Також виконаний аналіз напрямків удосконалення алгоритмів на основі мульти-нечіткого моделювання, сформовані комбіновані алгоритми обчислювальних процесів. В розділі вперше розроблені мульти-нечітка та реляційно-сепарабельна моделі процесів НЧПД і процедури її застосування, а також досліджена чисельна результативність, ефективність та точність алгоритмів. Проведені дослідження функціональної ефективності алгоритмів моделювання недетермінованих послідовностей з перемінним кроком, визначені особливості структури процесів моделювання числових послідовностей з перемінним кроком. В розділі подано опис структури засобів програмно-апаратного середовища для моделювання процесів НЧПД.

### 3. РОЗРОБКА СТРУКТУРИ ТА ІНСТРУМЕНТАЛЬНИХ ЗАСОБІВ ДЛЯ ДОСЛІДЖЕННЯ АЛОГОРИТМІВ МОДЕЛЮВАННЯ ЧИСЛОВИХ ПОСЛІДОВНОСТЕЙ З ПЕРЕМІННИМ КРОКОМ

#### 3.1.Формалізація завдань дослідження алгоритмів моделювання

Формалізація задачі представлена на рівні зовнішнього проектування наведена у вигляді діаграми варіантів використання. Користувач представлений у вигляді актора-дослідника, що взаємодіє із системою за допомогою варіантів використання. Варіанти використання надають опис можливостей, які система надає акторам.

Діаграма прецедентів визначає варіанти використання системи аналізу процесів моделювання ЧДП, що зображено на рис. 3.1

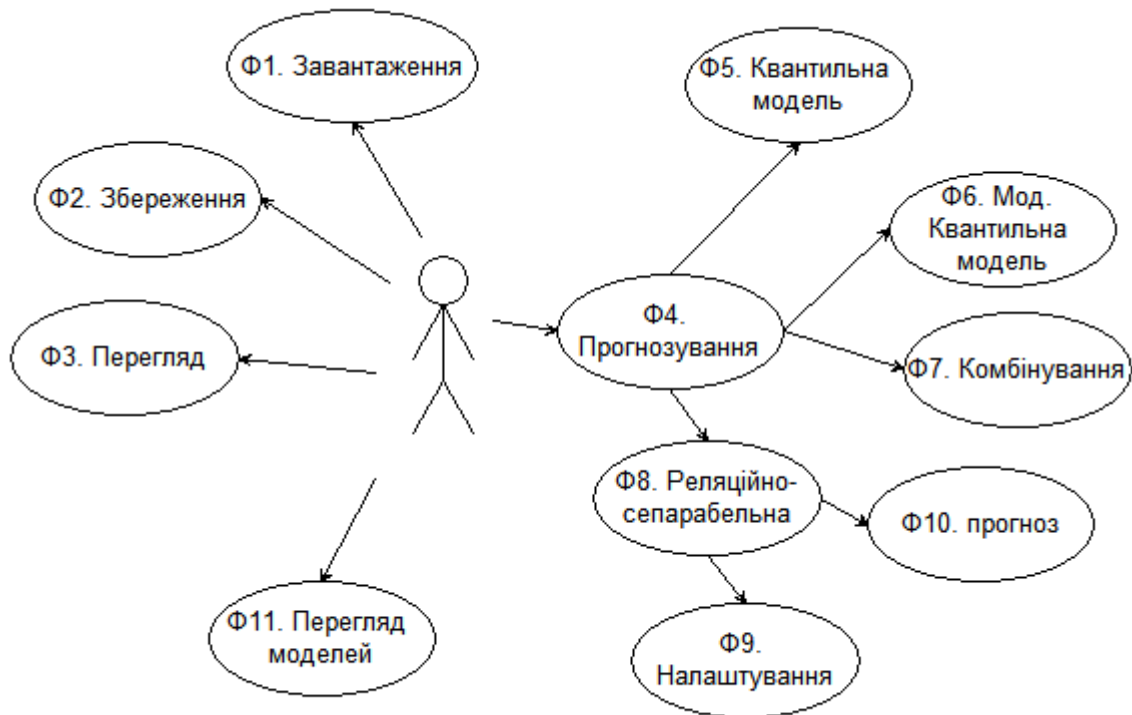


Рисунок 3.1 – Діаграма варіантів використання при дослідженні ЧП

На діаграмі представлені наступні функції програми: Ф1 – процедури отриманні вихідних даних для дослідження, як із текстового файлу, так і в форматі Excel; Ф2 – функція щодо збереження даних та результатів досліджень; Ф3 – функція перегляду завантажених даних; Ф3 – відображення графіків моделей процесів; Ф4 – вибір методу та налаштування

прогнозування; Ф5 – прогноз квантильним методом; Ф6 – прогноз модифікованим квантильним методом; Ф7 - прогноз комбінованим методом; Ф8 – прогноз реляційно-сепарабельним методом; Ф9 – налаштування для прогнозу; Ф10 - прогноз; Ф11 – перегляд побудованих моделей.

### 3.2. Розробка базової архітектури інструментальних засобів

В якості основної архітектури програмного комплексу "Дослідження методів моделювання часових послідовностей з нерівномірним кроком" використовується архітектура, побудована на основі інтеграції Visual Studio 2022, мови програмування C++ та технології Common Language Runtime (CLR). Visual Studio 2022 є потужним інтегрованим середовищем розробки, що надає розширені інструменти для створення програмного забезпечення на різних платформах.

Мова програмування C++ використовується для забезпечення високої продуктивності та ефективності в роботі з ресурсами системи. CLR є складовою частиною технологій .NET Framework та надає середовище виконання для програм, написаних на різних мовах програмування, що дозволяє їм взаємодіяти та використовувати об'єкти, створені на різних мовах.

Visual Studio 2022, C++ та CLR забезпечують розробникам широкий набір готових до використання інструментів та компонентів для реалізації різноманітних функціональностей, включаючи створення інтерфейсу користувача, роботу з базами даних, взаємодію з операційною системою та інші аспекти програмування.

Взаємодію головних елементів між частинами архітектурної моделі, представлено на рис. 3.2

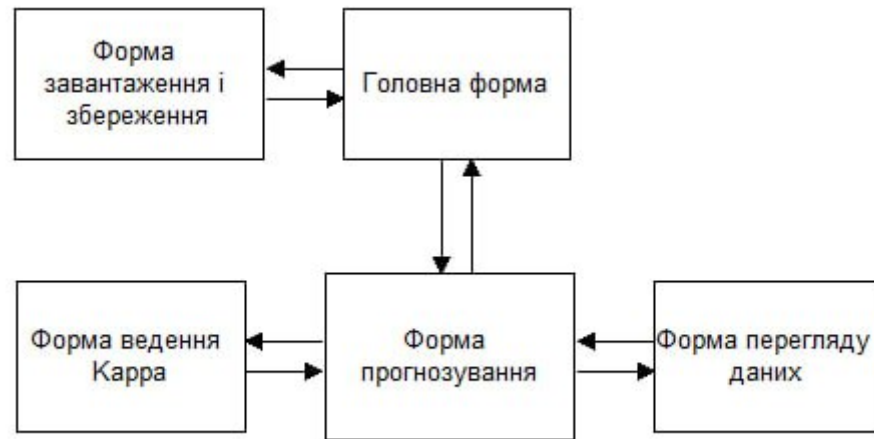


Рисунок 3.2 – Взаємодія головних елементів архітектурної моделі програми «Дослідження методів моделювання часових послідовностей з нерівномірним кроком».

### 3.3. Внутрішнє проектування інструментальних засобів

#### 3.3.1. Обґрунтування та вибір мови програмування

При виборі мови програмування урахувалися фактори стосовно можливості програмного середовища, а також можливостей безпосередньо мови програмування. Саме ці два головних критерії урахувалися при визначенні мови програмування.

Для розробки програмного комплексу було обране програмне середовище Visual Studio 2022 Professional.

При розробках програми «Дослідження сепарабельних моделей і FTS алгоритмів прогнозування процесів з нечіткими параметрами.» використано мову програмування C++, тому що данна мова має підтримку кількох парадигм програмування: об'єктно-орієнтованої, узагальненої, процедурної та ін. Цього достатньо при реалізації необхідних функцій для поставленого завдання та програмного комплексу загалом.

#### **Ієрархія та взаємодія класів програмної системи**

Взаємодія та проектування класів описано UML діаграмою.

Клас ForecastForm являється головним вікно програмного комплексу та породжує вікна KarraForm та ForecastStepForm. Розміщує часові ряди

компоненту Chart, керує таблицею даних TableLayoutPanel. Викликає допоміжні вікна.

Клас KappaForm являється допоміжним вікно програмного комплексу, необхідне для введення форми встановлення значення карра та графіки.

Клас ForecastStepForm являється допоміжним вікно програмного комплексу, необхідне для введення встановлення налаштування прогнозу та графіки.

Клас ForecastDS являється допоміжним класом, в якому зібрані основні розрахункові методи програмного комплексу. Опис класів зображено на рис. 3.3.

ForecastForm	ForecastDS
ForecastDS TestForecast; Chart form_graph; TableLavuoutPanel *form table;	vector<double> datasequence; struct_kappa_range kappa; struct_datasequence_range datasequence_range; struct_datasequence_range approx_scale_range; vector<double> approx_scale_3; vector<double> approx_scale_5; vector<double> forecast_multi; vector<double> forecast_source; vector<double> forecast_approx3; vector<double> forecast_approx5;
void UpdateGraph(); void UpdateTable(); void UpdateStatusStrip(); void UnlockKappa(); void UnlockForecast(); void UnlockForecastTest(); void UnlockDataView(); void UnlockForecastView(); void UnlockForecastTestView();	void defineDatasequence(vector<double> _datasequence); void defineKappa(double _value, double _range); void defineKappa(double _value_low, double _value_top, bool _range); void defineKappaAuto(double _value_low, double _value_top, double _step, bool _set); void defineDataqequenceRange(); vector<double> defineApproxScale(double min, double max, int number); void defineApproxScales(); void defineForecast(int _steps); double forecastMulti(double _m1, double _m2, double _m3, bool _save); double forecastSource(double _m1, double _m2, double _m3, bool _save); double forecastApprox3(double _m1, double _m2, double _m3, bool _save); double forecastApprox5(double _m1, double _m2, double _m3, bool _save); double forecastCalculateByMath(double _m1, double _m2, double _m3, double _left, double _right); double forecastCalculateByDed(double _m1, double _m2, double _m3, double _left, double _right); void testForecast();
KappaForm	
TextBox *value; TextBox *range; TextBox *final_range; Button *Save;	
void formkappa_save click(Object ^sender, EventArgs ^e) void formkappa_value_TextChanged(Object ^sender, EventArgs ^e); void formkappa_range_TextChanged(Object ^sender, EventArgs ^e)	
ForecastStepForm	
TextBox *value; Button *ForecastButton;	
void formforecast_click(Object ^sender, EventArgs ^e)	

Рисунок 3.3 – Класи програмного комплексу

Зв'язок між класами демонструє рис. 3.4.

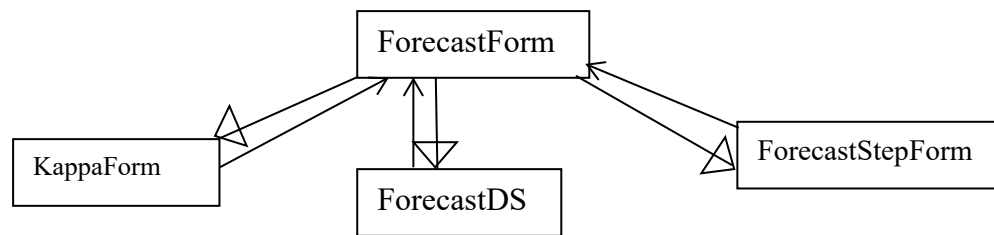


Рисунок 3.4 – Основні зв'язки між класами

### 3.3.2. Розробка архітектури ієрархії та взаємодії класів системи інструментальних засобів

При проектуванні внутрішньої структури системи використовувалися такі загальні принципи об'єктно-орієнтованого проектування:

- забезпечення слабкої зв'язності об'єктів що взаємодіють. За ним чим менше об'єкти знають один про одного, тим гнучкіше створювана програмна система. Кожному компоненту немає необхідності «знати» внутрішній устрій іншого;

- забезпечення взаємодії тільки з «близькими» компонентами.

Принцип спрямований на мінімальну інформованість – при проектуванні класу треба звертати увагу на кількість класів, з якими буде відбуватися взаємодія. Чим менше таких класів, тим гнучкіше система;

- голлівудський принцип – не викликайте нас, ми самі вас викличемо. За Фаулера - це синонім принципу IoC. Згідно ньому компоненти високого рівня (наприклад, інтерфейси) визначають за компоненти низького рівня (реалізації), як і коли їм підключатися до системи. Інші автори Head First Design Patterns допускають, що за цим принципом компоненти низького рівня можуть приймати участь в обчисленнях без формування залежностей з компонентами високого рівня. Саме в цьому полягає відмінність такого проектування від більш жорсткого принципу IoC;

- розділення інтерфейсу, за яким перевага віддається тому проекту, коли у системі використовується велика кількість спеціалізованих інтерфейсів замість кількох універсальних.

### 3.4. Формування інтерфейсу користувача програмного комплексу

При проектуванні інтерфейсу застосовується принцип інтуїтивної зрозумілості, при цьому інтерфейс був не нагромаджений великою кількістю елементів. Для цього було створено кілька форм, кожна з яких виконує свою роль.

Головна форма програмного комплексу зображена на рис. 3.5

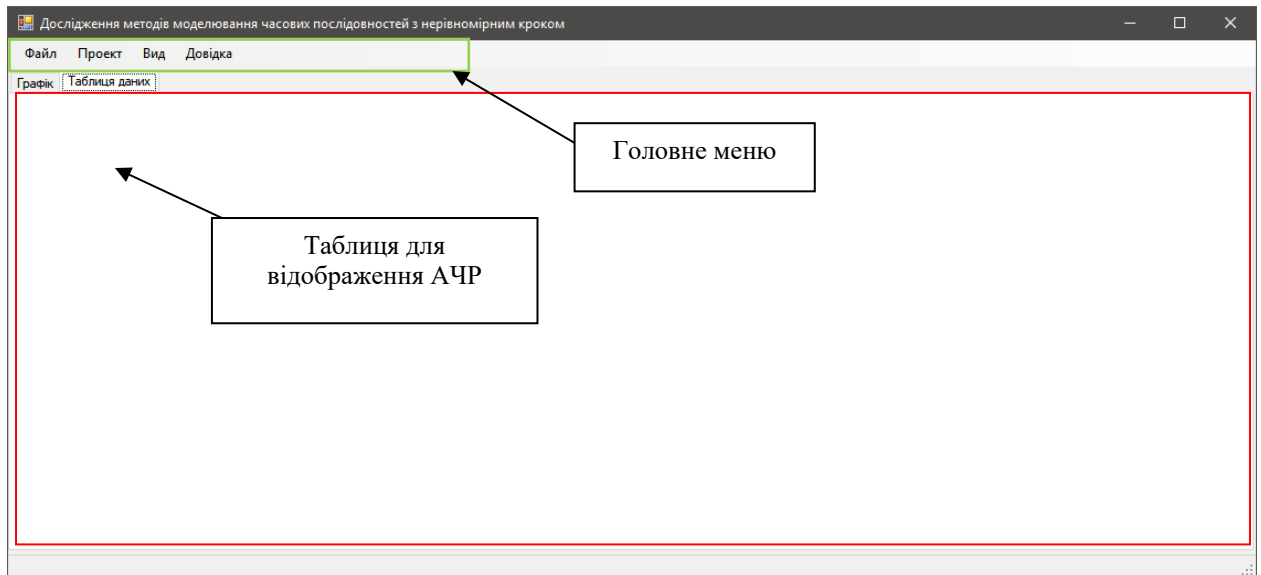


Рисунок 3.5 – Головне вікно програмного комплексу дослідження методів моделювання часових послідовностей з нерівномірним кроком

Вікно меню, призначене для введення параметру Карра зображена на рис. 3.6

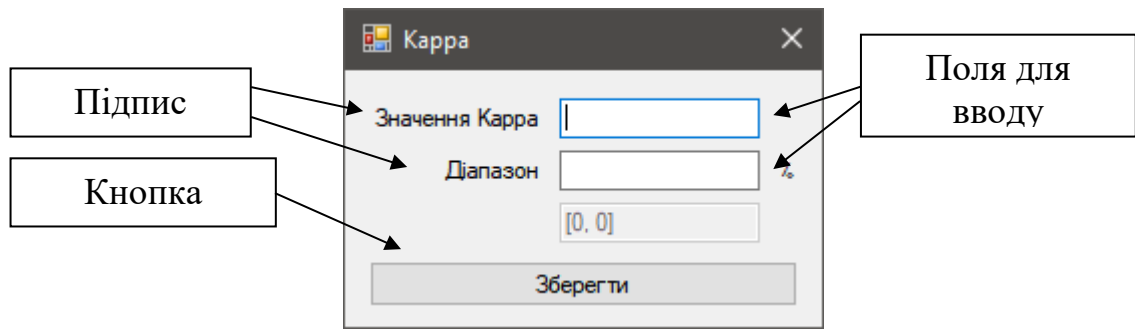


Рисунок 3.6 – Допоміжне вікно для введення параметру Карра

Вікно для відображення ряду на графіку зображено на рис. 3.7.

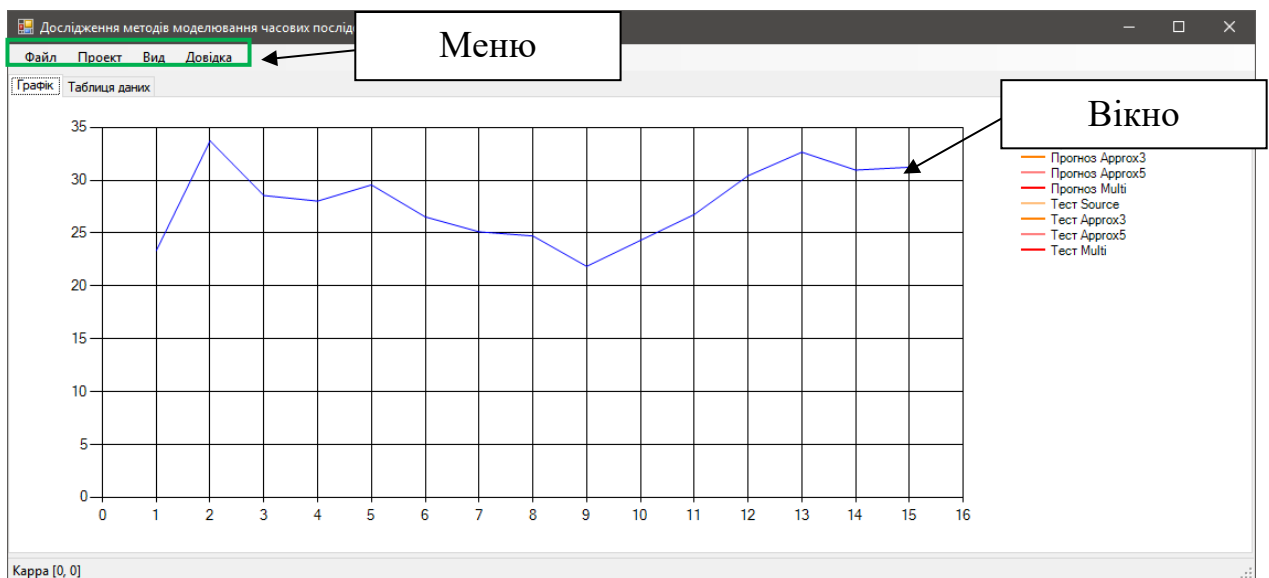


Рисунок 3.7 – Вікно для відображення графіку ЧР

### 3.5. Тестування та налагодження програми

#### 3.5.1. Аналіз та застосування методів тестування та налагодження

Етапи тестування та відлагодження програм є найбільш трудомісткими в процесах розробки програм. Мета тестування – це виявлення помилок у роботі програмного комплексу. Виявлення синтаксичних помилок в більшості виконується за допомогою компілятора, вони можуть виправляються ще на момент написання. Для виявлення логічних помилок, помилок типізації, помилок узгодженості параметрів площндур та ін. можна отримати лише дослідивши роботу алгоритму, шляхом порівняння очікуваних результати

роботи з спеціально сформованими вихідними даними щодо роботи алгоритму.

При виявленні логічної помилки її необхідно локалізувати та виправити. Для цього на практиці використовують процедури налагодження, коли знають в якому алгоритмі знаходиться локалізована логічна помилка. При відлагодженні перевіряється робота (виконання та дані) кожної команди, та команда алгоритму, яка не відповідає вставленим вимогам, і буде розглядатися як можлива логічна помилка..

Загальними та ефективними методами тестування алгоритму являються метод «чорної скриньки» та метод «білої скриньки».

При застосуванні методу «чорної скриньки» полягають що алгоритм певної функції невідомий, а відомі лише вхідні та очікувані вихідні данні функції, що підходить для виявлення логічної помилки у функції.

Метод «білої скриньки» передбачає що алгоритм реалізації функції відомий, а система вхідних даних будуються так, щоб покрити максимальну кількість гілок виконання алгоритму. Таким чином виявляється ділянка алгоритму, де можливо знаходиться логічна помилка. Після виявлення помилки починається етап відлагодження для виправлення помилки, з подальшим повторним тестуванням.

### Висновки до розділу 3

В результаті розробок за розділом виконана формалізація процесів програмного комплексу «Дослідження методів моделювання часових послідовностей з нерівномірним кроком» у вигляді діаграми варіантів. Діаграма варіантів представляє всі завдання, які необхідно реалізувати. Для формування базової архітектури програми використана архітектура бібліотеки.

Розроблено модель для відображення взаємодія між частинами архітектури програмного комплексу. У розділі наведено елементи щодо опису процесу проектування інтерфейсу користувача. Представлені методи тестування, виконане тестування фрагменту проограмного коду. Запропоновані та застосовані процедури розробки програмного забезпечення дозволяє зробити висновок про забезпечення достовірності результатів досліджень.

#### 4. ПОРІВНЯЛЬНІ ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ АЛГОРИТМІВ КЛАСИФІКАЦІЇ ТА МОДЕЛЮВАННЯ ЧИСЛОВИХ ПОСЛІДОВНОСТЕЙ З НЕРІВНОМІРНИМ КРОКОМ.

##### 4.1. Постановка завдань та підготовка даних експерименту

Метою дослідження являється створення моделей і засобів, призначених для моделювання та прогнозування процесів з недетермінованими і нечіткими параметрами. Зокрема, коли інтервал між окремими спостереженнями є нерівномірним і нечітким.

Аналіз літературних джерел показав незначну кількість таких досліджень. Разом з тим відзначається велика актуальність таких завдань.

В роботі в якості першоджерел таких процесів використані дані клінічного моніторингу, представлені на наступному слайді.

Відзначається нерівномірність кроку між контрольними заходами хворих, а також оцінювані показники фактично являються нечіткими .

Наприклад, рівень цукру в крові, тиск тощо. При проведенні повторних вимірів послідовно величини не однакові.

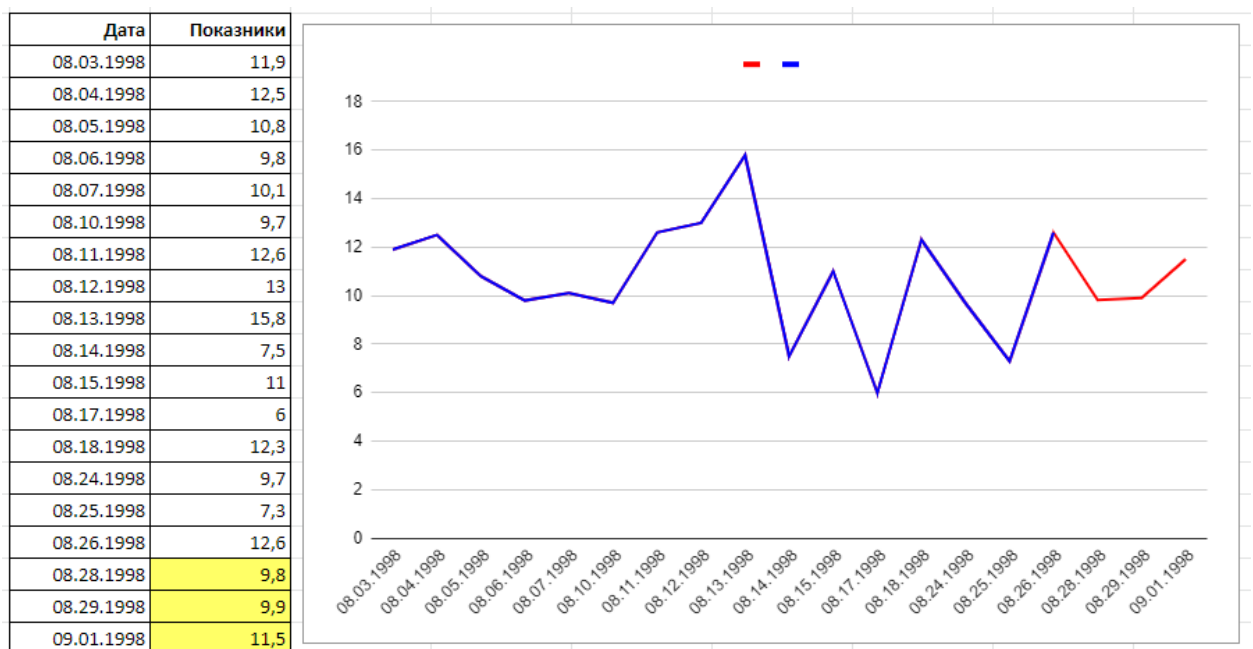


Рисунок 4.1 Представлення вихідних даних клінічного моніторингу пацієнта П1945

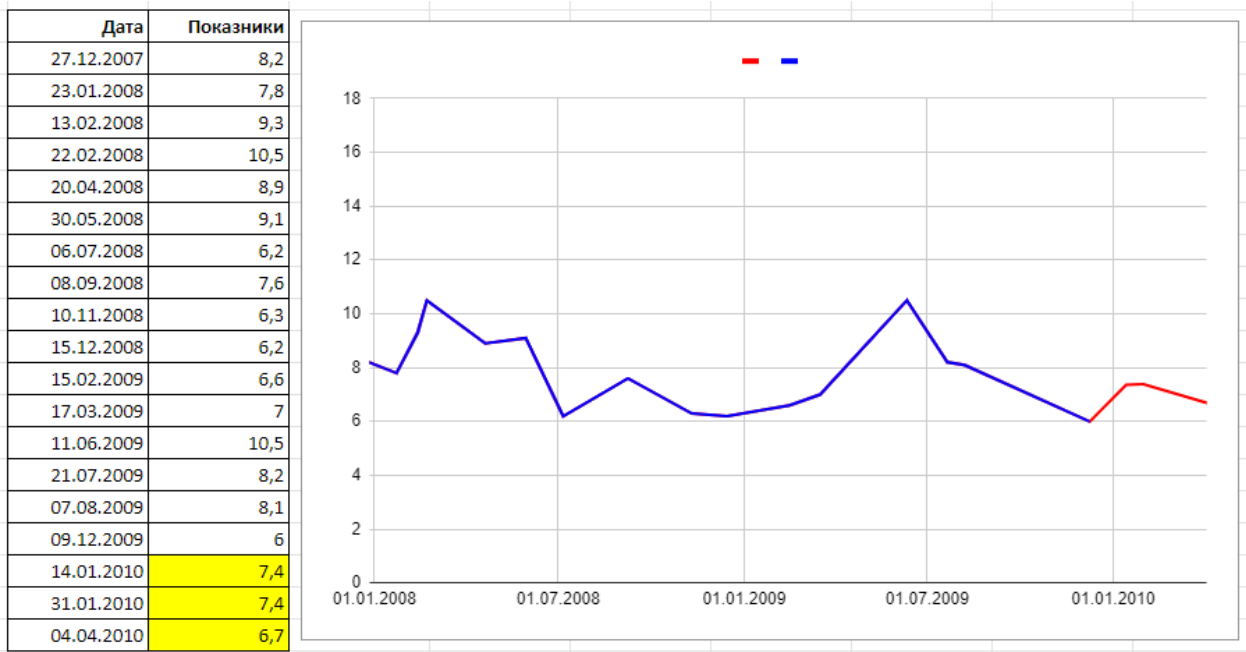


Рисунок 4.2 Представлення вихідних даних клінічного моніторингу пацієнта П1949

Метою з точки зору практики використання результатів оцінка періоду до наступного відвідування хворим лікаря, а також рівень його оцінюваних параметрів.

З метою моделювання недетермінованих та нечітких інтервалів застосовується сепарабельна модель, запропонована в роботах проф. Скалозуба В.В. та інших. При цьому виконується моделювання рівнів показників і інтервалів незалежно між собою.

Також в роботі запропонована нова, нечітко-реляційна модель для моделювання такого типу процесів. Досліджено можливості моделей 1-го та другого порядку. Що залежить від кількості попередніх етапів, які ураховуються при моделюванні та прогнозуванні.

Для забезпечення можливості моделювання таких коротких послідовностей (клінічних моніторинг конкретного хворого) в програмному комплексі спочатку дані розмножуються методом бутстреп. Він дозволяє отримати вибірки з більшим числом членів, із збереженням тих же властивостей процесу.

Далі виконуються процедури переходу від не відомих моделей інтервалів до нечіткого представлення цих послідовностей. Для цього в області можливих значень інтервалів водиться система нечітких величин, представлених на наступному слайді. Таким чином від довільних форм інтервалів переходимо до їх моделей в термінах Сі.

Для отриманих при цьому числових послідовностей виконується розрахунок прогнозного значення наступного інтервалу тощо. Таким чином для кожного  $\alpha$ -рівня знаходяться менша та більша оцінка прогнозного значення.

Засобами створюваного програмного комплексу передбачається виконати дослідження структури моделей нечітких величин, числа квантилів, нечітких величин, на які необхідно розділити області варіювання показників, при яких буде забезпечуватися достатня точність аналізу та прогнозування параметрів процесів, що спостерігаються.

#### 4.1.1. Опис структури засобів програмно-апаратного середовища

Методи моделювання та прогнозування були досліджено за допомогою розробленого програмного комплексу «Дослідження методів моделювання недетермінованих числових та нечітких послідовностей».

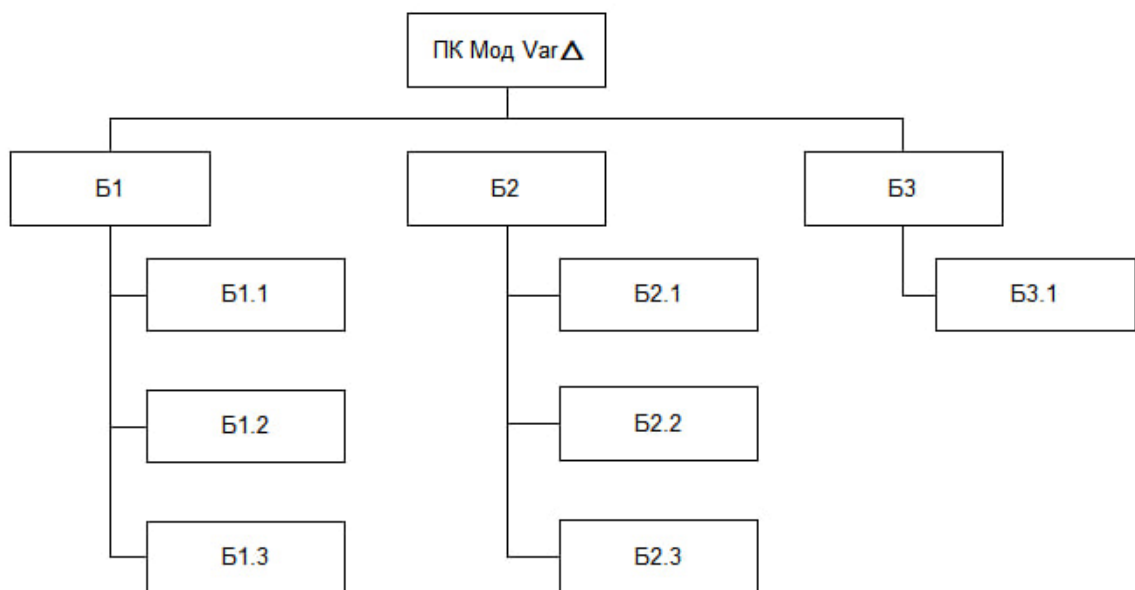


Рисунок 4.3 – Структура програмного комплексу аналізу та моделювання процесів представлених нечітким кроком спостережень.

На рисунку Рис. 4.3 наведена загальна структура програмного комплексу моделювання процесів, представлених спостереженнями з нерівномірним кроком між спостереженнями. При цьому блоки визначають наступне:

Б1 – Комплекс функцій щодо забезпечення управління даними; Б1.1 – Отримання даних досліджуваного процесу, представленого послідовністю дійсних чисел  $(H)$ , де – послідовність « $H$ » визначається величинами показника, упорядкованими за номерами спостережень  $(h_1, h_2, \dots, h_k, \dots, h_n)$ ; Б1.2 – збереження, контроль, формування представлення даних  $(H)$ ; Б1.3 – функції щодо відображення даних процесів  $(H)$ ; Б2 – комплекс функцій із моделювання і прогнозування значень показників  $(H)$ ; Б2.1 – аналіз та прогнозування  $(H)$ , за алгоритмом квантильної моделі процесу; Б2.2 – аналіз та прогнозування  $(H)$ , за алгоритмом модифікованої квантильної моделі процесу; Б2.3 – аналіз та прогнозування  $(H)$ , за алгоритмом комбінування моделей процесу; Б.3 - комплекс функцій із відображення результатів моделювання і прогнозування значень показників  $(H)$ ; Б3.1 – відображення даних для моделювання  $(H)$ .

#### 4.1.2. Алгоритми та процедури методу моделювання нерівномірних часових послідовностей

З метою дослідження методів моделювання часових послідовностей з нерівномірним кроком застосовується сепарабельна модель, запропонована в роботах проф. Скалозуба В.В. та інших. При цьому виконується моделювання рівнів показників і інтервалів незалежно між собою.

На рис. 4.4 приведена загальна структура класичного квантильного алгоритму (1.2) – (1.7), яка відображає процедуру визначення тренду процесу, який досліджується, а також розрахунку прогнозних значень на наступному кроці, новий рівень процесу, що враховує показник «тренд».

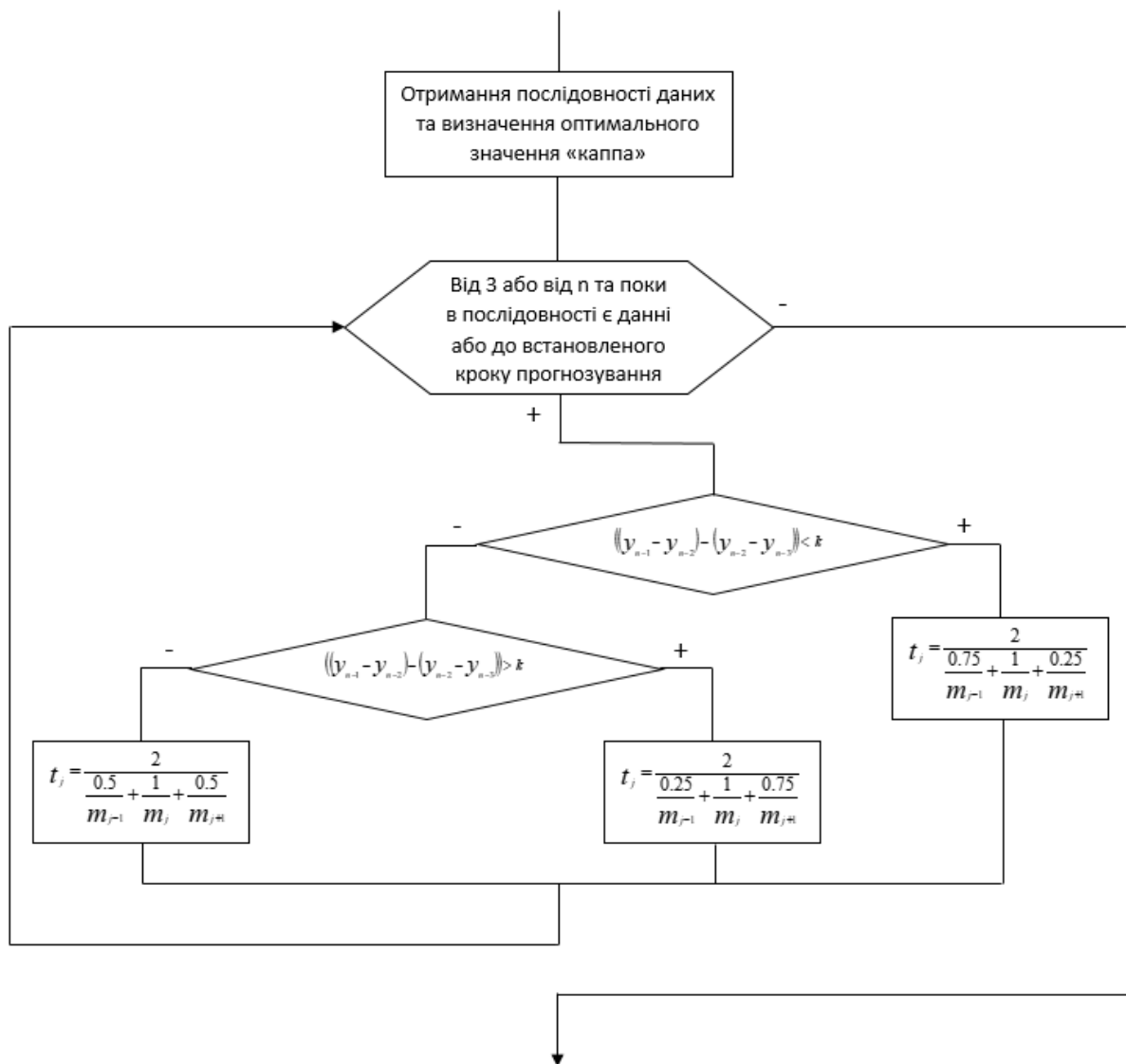


Рисунок 4.4 – Блок-схема алгоритму прогнозування.

#### 4.1.3. Опис методики відбору даних для експериментальних досліджень алгоритмів

В роботі в якості першоджерел таких процесів використані дані клінічного моніторингу, представлені на наступному слайді. Відзначається нерівномірність кроку між контрольними заходами хворих, а також оцінювані показники фактично являються нечіткими. Наприклад, рівень цукру в крові, тиск тощо. При проведенні повторних вимірів послідовно величини не однакові.

## 4.2. Процедура та результати проведення експериментальних досліджень алгоритмів моделювання нерівномірних часових послідовностей

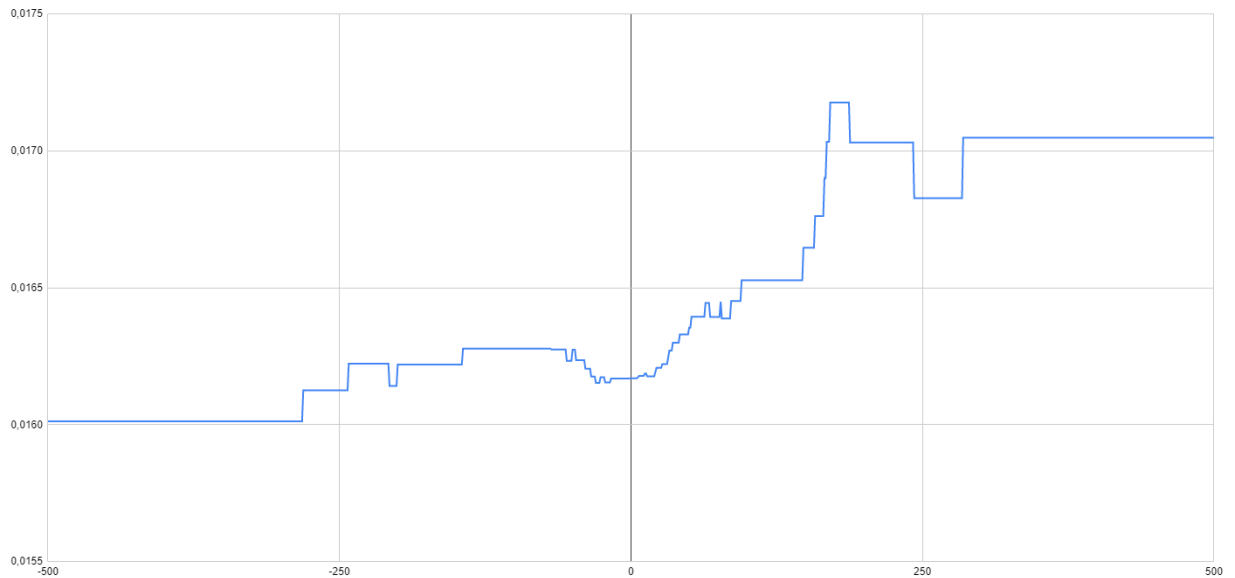


Рисунок 4.5 – Середня похибка для запропонованого алгоритму

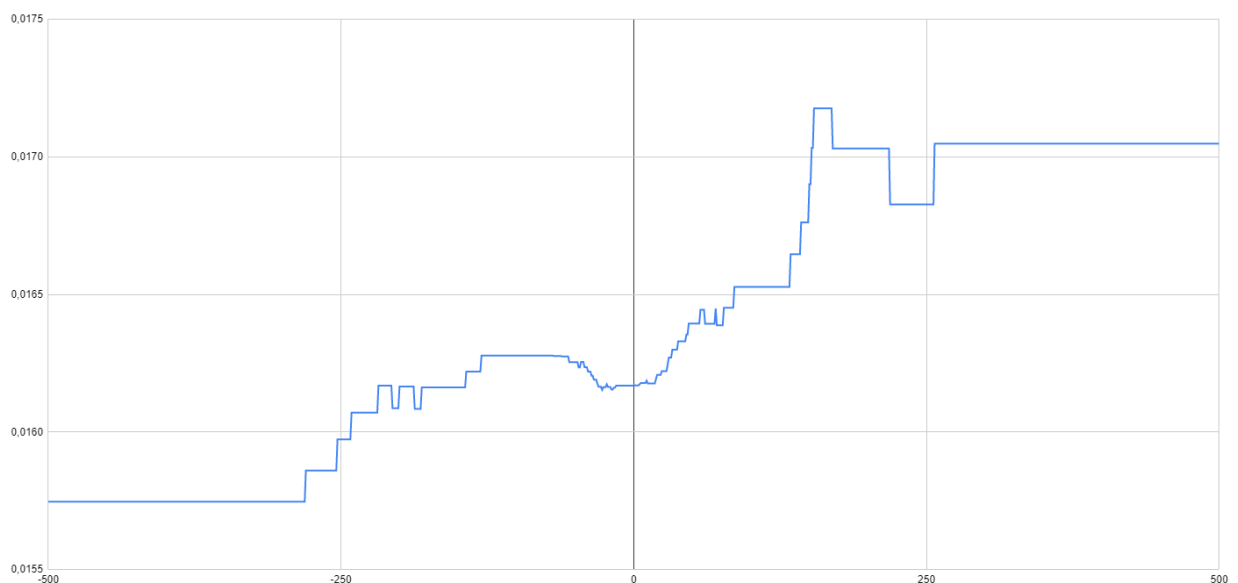


Рисунок 4.6 – Середня похибка для удосконаленого алгоритму

В експериментальних розрахунках значень показників інтерполяції рівнів, отриманих за моделями процесів моделювання часових послідовностей з нерівномірним кроком.

Експерименти з дослідження методу класифікації проводились за допомогою застосування розробленого програмного комплексу

На рисунках рис. 4.5 та рис. 4.6 наведено результати застосування процедур інтерполяції внутрішньо інтервальних рівні моделей часових послідовностей з нерівномірним кроком.

Таблиця 4.1

## Порівняльний аналіз точності алгоритмів

Початкові данні	<b>28.02</b>	<b>29.56</b>	<b>26.53</b>	<b>25.12</b>	<b>24.73</b>	<b>21.85</b>	<b>24.31</b>	<b>26.74</b>
Розрахунки за квантильним алгоритмом	<b>27.08</b>	<b>30.68</b>	<b>28.85</b>	<b>28.38</b>	<b>27.36</b>	<b>25.65</b>	<b>24.52</b>	<b>24.18</b>
Розрахунки за комбінованим алгоритмом	<b>27.55</b>	<b>29.18</b>	<b>28.53</b>	<b>28.35</b>	<b>26.44</b>	<b>25.00</b>	<b>24.35</b>	<b>23.90</b>
Відносна точність у відсотках	<b>1.68%</b>	<b>1.29%</b>	<b>7.54%</b>	<b>12.86%</b>	<b>6.91%</b>	<b>14.42%</b>	<b>0.16%</b>	<b>10.62%</b>

Таблиця 4.1 містить порівняльний аналіз точності запропонованих алгоритмів, в якому подано відносну точність у відсотках в порівнянні з комбінованим алгоритмом та вхідними даними. Визначення відносної точності проводилося на основі результатів комбінованого алгоритму та їх приближення до вхідних даних. Максимальна похибка становила 12.9%, що вважається прийнятною через невелику кількість рівнів у ряду.

#### Висновки до розділу 4

У розділі досліджено методи моделювання та прогнозування часових послідовностей з нерівномірним кроком. При цьому порівнювалися алгоритми FTS першого порядку на основі нечітких реляційних відношень, метод FTS третього порядку з трендом, а також процедура на основі мульти-нечіткого алгоритму, яка використовувала метод FTS третього порядку з трендом, який використовував два представлення (дві апроксимації) області варіювання процесу. А саме, для 5 та 3-х нечітких трикутних моделей квантилів. Отримані при цьому результати розрахунків узагальнювалися за рахунок вагових коефіцієнтів.

Отримані результати свідчать про коректність запропонованих розробок математичних моделей, алгоритмів і програмних засобів, що дозволило підвищити точність процедур аналізу та прогнозування процесів з нерівномірним кроком між спостереженнями.

Розроблений програмний комплекс для моделювання та прогнозування забезпечує функції дослідження процесів з нерівномірними інтервалами спостережень.

## ВИСНОВКИ

Дипломна робота «Дослідження методів моделювання часових послідовностей з нерівномірним кроком» присвячена розробкам та дослідженню завдань щодо формування засобів аналізу та моделювання часових послідовностей з перемінним (недетермінованим) кроком спостережень. Актуальність роботи визначається тим, що дослідження характеристик складних процесів і систем потребує розвитку засоби щодо ефективного та достовірного застосування процедур комп'ютерне моделювання на основі числових рядів спостережень даних складних технологічних об'єктів. Натепер моделі та методи аналізу та моделювання часових рядів (ЧР) з перемінним кроком спостережень є не достатньо дослідженими. Тому завдання щодо розвитку методів та засобів моделювання нерівномірних у часі ЧР досліджені не в повній мірі, має науковий і практичний інтерес.

Об'єктом дослідження дипломної роботи були процеси моделювання параметрів часових послідовностей, отриманих з нерівномірним кроком, які являються, в проведених розрахунках зокрема, часовими рядами процесів моніторингу станів хворих. А предметом дослідження були математичні моделі, алгоритми та програмні процедури, призначені для аналізу та прогнозування нерівномірних у часі недетермінованих числових послідовностей. Метою досліджень та розробок було забезпечення можливостей щодо підвищення ефективності та точності методів і алгоритмів аналізу та прогнозування характеристик ЧР з нерівномірним кроком. В якості методів, що використовувалися в роботі, були використані методи системного аналізу та аналізу даних і часових рядів, методи математичного моделювання і математичної статистики та порівняльного аналізу, методи нечіткого, а також комп'ютерного моделювання. В роботі використані методи та технології щодо проєктування та створення програмних систем, методи алгоритмізації; експериментального дослідження програмного забезпечення.

В доплонній роботі були реалізовані такі завдання

- розроблення алгоритмічних та програмних засобів і зробити порівняльний аналіз щодо числової ефективності алгоритмів моделювання ЧР з нерівномірним кроком спостережень.

- розробити алгоритмічні та програмні засоби та виконати прогнозування часових послідовностей з нерівномірним кроком спостережень,

- розробити інструментальне програмне середовище, яке забезпечує дослідження алгоритмів моделювання нерівномірних у часі недетермінованих числових послідовностей;

- виконати порівняльний аналіз алгоритмів моделювання нерівномірних у часі послідовностей щодо точності та чисельної ефективності;

- отримати рекомендації стосовно застосування процедур моделювання недетермінованих послідовностей з нерівномірним кроком спостережень.

#### Наукова новизна результатів роботи

- виконано порівняльний аналіз чисельної ефективності алгоритмів моделювання та прогнозування характеристик ЧР з перемінним кроком спостережень,

- розроблено мульти-нечітку реліційно-сепарабельну модель прогнозування послідовностей з перемінним кроком,

- удосконалено процедуру формування комбінованих алгоритмів моделювання досліджуваних процесів, що забезпечує коректність результатів комп'ютерного моделювання та підвищення точності показників процесів.

Практичне значення мають результати проведеного аналізу стану щодо застосування моделей та алгоритмів моделювання та прогнозування ЧР, що мають нерівномірний крок спостережень, результати порівняльного аналізу чисельної ефективності алгоритмів моделювання та прогнозування характеристик числових послідовностей з перемінним кроком спостережень, результати застосування процедур формування мульти-нечітких та комбінованих алгоритмів моделювання досліджуваних процесів, що забезпечує коректність результатів комп'ютерного моделювання та підвищення точності показників процесів, а також програмні засоби для

модельовання та прогнозування характеристик числових послідовностей з перемінним кроком спостережень.

Результати дипломної роботи мають апробацію. Результати доповідались на семінарах кафедри КІТ 30.11.2022 р. та 11.01.2024 р. Матеріали роботи доповідалися на 16 та 17 міжнародних науково-практичних конференціях «Сучасні інформаційні та комунікаційні технології на транспорті, в промисловості та освіті» (14 – 15 грудня 2022 р. та 13-14 грудня 2023 р..) м. Дніпро, Український державний університет науки і технологій, що приведено у додатках.

## СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Прогнозування і моделювання (Oracle Crystal Ball) [Електронний ресурс] – Режим доступу: [http://ubc-corp.ru/ru/oracle\\_crystal\\_ball](http://ubc-corp.ru/ru/oracle_crystal_ball).
2. Прогнозування в Predictive Planning (Oracle Crystal Ball) [Електронний ресурс] – Режим доступу: <https://ivan-shamaev.ru/predictive-planning-oracle-crystal-ball/>
3. Tahseen A., Aqil S., Burney Cemal A. A New Quantile Based Fuzzy Time Series Forecasting Model [Електронний ресурс] – Режим доступу: <https://publications.waset.org/14214/pdf>
4. Qiang S., Brad S., Forecasting enrollments with fuzzy time series - Part I [Електронний ресурс] – Режим доступу: <https://www.sciencedirect.com/science/article/pii/016501149390355L>
5. Sheng T., Yi-Chung C. Deterministic fuzzy time series model for forecasting enrollments [Електронний ресурс] – Режим доступу: <https://www.sciencedirect.com/science/article/pii/S0898122107001630>
6. Tahseen A., Jilani S., Aqil B., Ardil C. Multivariate High Order Fuzzy Time Series Forecasting for Car Road Accidents [Електронний ресурс] – Режим доступу: [https://www.researchgate.net/publication/285870449\\_Multivariate\\_High\\_Order\\_Fuzzy\\_Time\\_Series\\_Forecasting\\_for\\_Car\\_Road\\_Accidents](https://www.researchgate.net/publication/285870449_Multivariate_High_Order_Fuzzy_Time_Series_Forecasting_for_Car_Road_Accidents)
7. Q. Song, and B. S. Chissom, “Forecasting enrollments with fuzzy time series — Part I,” Fuzzy Sets and Systems, vol. 54, issue 1, 1993a, pp. 1–9.
8. B. S. Chissom, “Fuzzy time series and its models,” Fuzzy Sets and Systems, vol. 54, issue 3, 1993b, pp. 269-277.
9. Q. Song, and B. S. Chissom, “Forecasting enrollments with fuzzy time series — Part II,” Fuzzy Sets and Systems, vol. 62, 1994, pp. 1-8.
10. S. M. Chen, “Forecasting enrollments based on fuzzy time series,” Fuzzy Sets and Systems, vol. 81, 1996, pp. 311-319.

- 11.Q. Song, "A note on fuzzy time series model selection with sample autocorrelation functions," *Cybernetics and Systems: An International Journal*, vol. 34, 2003, pp. 93-107.
- 12.Q. Song, and R.P. Leland, "Adaptive learning defuzzification techniques and applications," *Fuzzy Sets and Systems*, vol. 81, 1996, pp. 321-329.
- 13.J. R. Hwang, S. M. Chen, and C. H. Lee, "Handling forecasting problems using fuzzy time series", *Fuzzy Sets and Systems*, vol. 100, 1998, pp.217-228.
- 14.K. Huarng, "Heuristic models of fuzzy time series for forecasting," *Fuzzy Sets and Systems*, vol. 123, issue 3, 2001a, pp. 369-386
- 15.K. Huarng, "Effective lengths of intervals to improve forecasting in fuzzy time series," *Fuzzy Sets and Systems*, vol. 123, issue 3, 2001b, pp. 387-394.
- 16.S.-M. Chen, "Forecasting enrollments based on high-order fuzzy time series," *Cybernetics and Systems: An International Journal*, vol. 33, pp. 1-16.
- 17.О. Мулеса, В. Сницьюк та С. Герзанич, «Метод нечіткої класифікації на основі послідовного аналізу вальда,» *Automation of technological and business processes*, т. №11, pp. 35-42, 2020.
- 18.Скалозуб В.В. Дослідження алгоритмів оптимального формування транспортно-логістичних процесів реального часу гетерогенними операторами / Скалозуб В.В., Горячкін В.М., Скалозуб М.В., Терлецький І.А. // Транспортні системи та технології перевезень. Збірник наукових праць ДНУЗТ ім. акад. В. Лазаряна. – Вип. 23. – 2022 р. – С. 65-73. <https://doi.org/10.15802/tstt2022/261657>
- 19.Скалозуб В.В. Дослідження процедур мережі Хеммінга для управління сервісними системами при неточно визначених і природомовних даних / Скалозуб В.В., Горячкін В.М., Клименко І.В., Терлецький І.А., Терленко А.П. // Наука та прогрес транспорту. – 2022. – № 3-4 (99-100). – С. 33-47. <https://doi.org/10.15802/stp2022/276411>
- 20.Скалозуб В.В., Горячкін В.М., Мурашов О.В. РЕЛЯЦІЙНО-СЕПАРАБЕЛЬНІ МОДЕЛІ ПРОЦЕСІВ МОНІТОРИНГУ ПРИ ПЕРЕМІННИХ І НЕЧІТКИХ ІНТЕРВАЛАХ СПОСТЕРЕЖЕНЬ // Системні

технології. 2023. Регіонального міжвузівського збірника наукових праць «Системні технології». Випуск 4 (147) , 2023. С. 4-20.

21. Скалозуб В.В., Васенін М.О. ПИТАННЯ ФОРМУВАННЯ МУЛЬТИ-НЕЧІТКИХ МОДЕЛЕЙ І ПРОЦЕДУР СИСТЕМ МОНІТОРИНГУ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ // Інформаційні технології в металургії та машинобудуванні. ІТММ'2023: тези доповідей міжнародної науково-практичної конференції (Дніпро, 22 березня 2023 р.) . – Дніпро: УДУНТ, 2023. С. 182 – 183

DOI: 10.34185/1991-7848.itmm.2023.01.050

22. Скалозуб В.В., Горячкін В.М., Мурашов О.В. РЕЛЯЦІЙНО-СЕПАРАБЕЛЬНІ МОДЕЛІ ПРОЦЕСІВ МОНІТОРИНГУ ПРИ ПЕРЕМІННИХ І НЕЧІТКИХ ІНТЕРВАЛАХ СПОСТЕРЕЖЕНЬ // Інформаційні технології в металургії та машинобудуванні. ІТММ'2023: тези доповідей міжнародної науково-практичної конференції (Дніпро, 22 березня 2023 р.) . – Дніпро: УДУНТ, 2023. С. 184 – 187.

DOI: 10.34185/1991-7848.itmm.2023.01.051

23. Мурашов О.В. РОЗВИТОК МОДЕЛЕЙ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ МОНІТОРИНГУ ПРИ НЕРІВНОМІРНИХ І НЕЧІТКИХ ІНТЕРВАЛАХ СПОСТЕРЕЖЕНЬ // СУЧАСНІ ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ТА СИСТЕМИ В УПРАВЛІННІ Збірник матеріалів ІV Міжнародної науково-практичної конференції молодих вчених, аспірантів і студентів (Інтернет-конференція) 20–21 квітня 2023 р. Київ, КНЕУ. - С 152.

DOI: 10.34185/1991-7848.itmm.2023.01.051

24. Мурашов О.В. РОЗВИТОК МОДЕЛЕЙ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ МОНІТОРИНГУ ПРИ НЕРІВНОМІРНИХ І НЕЧІТКИХ ІНТЕРВАЛАХ СПОСТЕРЕЖЕНЬ // СУЧАСНІ ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ТА СИСТЕМИ В УПРАВЛІННІ Збірник матеріалів ІV Міжнародної науково-практичної конференції молодих вчених, аспірантів

- і студентів (Інтернет-конференція) 20–21 квітня 2023 р. Київ, КНЕУ. - С 152.
25. Mahua Bose, Kalyani Mali. Designing fuzzy time series forecasting models: A survey // International Journal of Approximate Reasoning, Volume 111, August 2019, Pages 78-99. <https://doi.org/10.1016/j.ijar.2019.05.002>
26. Mahua Bose, Kalyani Mali. Forecasting with Multivariate Fuzzy Time Series: A Statistical Approach Published 7 September 2019 Computer Science. DOI:10.1007/978-981-15-1366-4\_20. International Conference on Machine Intelligence and Signal Processing MISP 2019: Machine Intelligence and Signal Processing pp 247–257
27. Ming-Tao Chou. Fuzzy Forecast Based on Fuzzy Time Series DOI: <http://dx.doi.org/10.5772/intechopen.82843>
28. Mona Mahmoud Samy Abo El Nasr Fuzzy Time Series Forecasting: Chen, Markov Chain and Cheng Models1 Alexandria Univesity for Administrative Sciences© – Vol. 60 – No. 2 – March 2023. P.33-45. DOI:10.21608/acj.2023.294123
29. Patricia O. Lucas, Omid Orang, Petronio C., Eduardo M. A Tutorial on Fuzzy Time Series Forecasting Models: Recent Advances and Challenges. Learning and Nonlinear Models. Vol. 19. 2022. P.29-50. DOI:10.21528/lnlm-vol19-no2-art3
30. Скалозуб В.В., Білий Б.Б., Галабут О.О., Мурашов О.В. Методи інтелектуального моделювання процесів з перемінним інтервалом спостережень та конструктивного упорядкування «з вагою». Системные технологии. Региональный межвузовский сборник научных работ. Выпуск 3 (128), 2020. С.127-143. DOI: 10.15802/tstt2020/208705
31. Скалозуб В.В., Горячкін В.М., Мурашов О.В. Комплексні моделі упорядкування мульти-послідовностей з нечіткими параметрами. Наука та прогрес транспорту. Вісник Дніпропетровського національного університету залізничного транспорту імені академіка В. Лазаряна. Выпуск 2 (92). 2021. С. 135-150. DOI:10.15802/stp2021/237291

32. Скалозуб В.В., Мурашов О.В. Моделювання даних процесів моніторингу при нерівномірних і нечітких інтервалах спостережень. «Системні технології». Випуск 4 (135). 2020. С.135-148. DOI 10.34185/1562-9945-4-135-2021-14

## **ДОДАТКИ**

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ЗАТВЕРДЖУЮ  
Перший проректор УДУНТ

А.В. Радкевич

«Дослідження методів моделювання часових  
послідовностей з нерівномірним кроком»

Технічне завдання  
ЛИСТ ЗАТВЕРДЖЕННЯ  
44165850.1218-01-ЛЗ

Завідувач кафедри КІТ

доц. \_\_\_\_\_ Горячкін В.М.  
(посада) (підпис) (ПІБ)

Керівник дипломної роботи

проф. \_\_\_\_\_ Скалозуб В.В.  
(посада) (підпис) (ПІБ)

Виконавець Студент гр. ПЗ2221

\_\_\_\_\_ Старина А.К.

Нормоконтролер:

\_\_\_\_\_ Світлана ВОЛКОВА

«Дослідження методів моделювання часових  
послідовностей з нерівномірним кроком»

Технічне завдання

44165850.1218-01

Аркушів 24

3  
44165850.1218-01  
АНОТАЦІЯ

Документ 44165850.1218-01 «Дослідження методів моделювання часових послідовностей з нерівномірним кроком» Технічне завдання, що входить до складу програмної документації до дипломного проекту.

У даному документі представлено призначення та область застосування програми, основні вимоги, стадії та строки виконання проекту, технічні та техніко-економічні показники, що пред'являються до програми.

ЗМІСТ

ВСТУП.....	4
1 ПІДСТАВА ДЛЯ РОЗРОБКИ .....	5
2 ПРИЗНАЧЕННЯ РОЗРОБКИ.....	6
2.1 Функціональне призначення .....	6
2.2 Експлуатаційне призначення .....	6
3 ВИМОГИ ДО ПРОГРАМИ .....	7
3.1 Вимоги до функціональних характеристик.....	7
3.2 Вимоги до надійності.....	7
3.3 Умови експлуатації .....	7
3.4 Вимоги до складу та параметрів технічних засобів .....	8
3.5 Вимоги до інформаційної та програмної сумісності.....	8
3.6 Вимоги до маркування та пакування .....	8
3.7 Вимоги до транспортування та зберігання.....	<b>Помилка! Закладку не визначено.</b>
4 ВИМОГИ ДО ПРОГРАМНОЇ ДОКУМЕНТАЦІЇ.....	10
5 КОШТОРИС НА РОЗРОБКУ ПЗ .....	11
6 СТАДІЇ ТА ЕТАПИ РОЗРОБКИ.....	21
7 ПОРЯДОК КОНТРОЛЮ ТА ПРИЙОМУ.....	22
СПИСОК ЛІТЕРАТУРИ.....	23

«Дослідження методів моделювання часових послідовностей з нерівномірним кроком» – це програмний комплекс для аналізу та дослідження властивостей складних процесів, які виявляються в різноманітних технологічних, виробничих, інформаційних, лікувальних та інших системах. Розглядаючи такі процеси, які можуть мати значний ступінь невизначеності та нерівномірності в часі, важливо зазначити, що їх аналіз та моделювання вимагають ефективних і надійних методів.

Різнманітні технологічні, виробничі та інші системи можуть функціонувати у складних умовах, де часові послідовності їхніх характеристик можуть змінюватися нерегулярно. При цьому для отримання вихідних даних для моделювання можливо використовувати лише часові ряди, які представляють собою послідовність значень, вимірюваних в різні моменти часу.

У якості прикладів таких об'єктів досліджень розглянуто різні сфери, включаючи технологічні процеси виробництва, системи інформаційного моніторингу, а також динаміку лікувальних процедур. Проектування та розробка програмного забезпечення для ефективного аналізу та моделювання цих часових послідовностей стає важливим завданням у високотехнологічному середовищі.

Такий комплекс досліджень відкриває можливості для вдосконалення алгоритмів та методів, спрямованих на аналіз та прогнозування характеристик недетермінованих числових послідовностей з нерівномірним кроком спостережень.

**1 ПІДСТАВА ДЛЯ РОЗРОБКИ**

Основою для розробки є наказ ректора Українського державного університету науки і технології Радкевич А.В. «Про затвердження тем та призначення керівників дипломних проєктів» № 88888 ст від 111111.202111р. «Про призначення керівників та затвердження тем магістерських робіт» факультету «Комп'ютерні технології і системи» за спеціальністю 121 «Інженерія програмного забезпечення» по кафедрі «Комп'ютерні інформаційні технології».

У відповідності з наказом, тема дипломного проєкту – «Дослідження методів моделювання часових послідовностей з нерівномірним кроком» Керівник проєкту проф. Скалозуб В. В.

## 2 ПРИЗНАЧЕННЯ РОЗРОБКИ

### 2.1 Функціональне призначення

Програмний комплекс призначений для дослідження методів моделювання часових послідовностей з нерівномірним кроком.

Функціональним призначенням програми є наступне:

- побудова прогнозів на основі початкових даних з нерівномірним кроком;
- побудова;
- розмноження

### 2.2 Експлуатаційне призначення

Експлуатаційне призначення програмного комплексу полягає в прогнозуванні наступних значень в послідовності даних.

. Експлуатаційними параметрами порівняння точності, які визначаються за призначенням програмного комплексу, є:

- порівняння;
- прогнозування;

## **3 ВИМОГИ ДО ПРОГРАМИ**

### **3.1 Вимоги до функціональних характеристик**

Програма повинна виконувати прогноз на основі послідовності даних з файлу та відображати їх на графіку:

- Відображення графіку;
- Розрахунок прогнозованих значень;
- Відображення прогнозів.

#### **Вхідні дані:**

- Послідовність дійсних чисел та значення карра.

Вхідні дані надаються користувачем шляхом загрузки картинки з комп'ютера користувача, або за допомогою графічного редактора, який вбудований у програму.

#### **Вихідні дані:**

- Послідовність дійсних чисел, що являють собою прогноз значень.

### **3.2 Вимоги до надійності**

Для забезпечення надійного функціонування необхідно забезпечити наявність архівного коду тексту програми на зовнішньому носії.

### **3.3 Умови експлуатації**

Для забезпечення надійної роботи програмного продукту користувачу необхідно дотримуватись таких умов:

- програмний засіб повинен використовуватись у приміщеннях, призначених для роботи з ЕОМ з відповідними кліматичними умовами;
- працювати з програмним засобом може людина, що має навички роботи з ПК та ознайомена з посібником користувача.

### **3.4 Вимоги до складу і параметрів технічних засобів**

Програмний продукт розрахований для функціонування на пристроях, що мають такі мінімальні характеристики:

- процесор: Intel Core 2 Duo;
- місце на диску: 512 Мб;
- об'єм оперативної пам'яті: 2 Gb;
- маніпулятори: клавіатура, комп'ютерна «миша»;
- інше устаткування: монітор.

### 3.5 Вимоги до маркування і упаковки

Упаковка програмного комплексу, включаючи документацію, повинна бути захищена від пошкоджень різного роду (механічних, кліматичних).

На упаковці повинно бути вказано назву програмного комплексу, номер версії, якщо вона змінювалась, мінімальні системні вимоги .

На зворотній стороні упаковки вказується розробник та його юридична адреса. На рис. 3.1 приведений приклад маркування.

«Дослідження методів моделювання часових  
послідовностей з нерівномірним кроком» версія 1.0  
Розробник: Старина Андрій  
Кафедра «КІТ» ДНУЗТ  
2024

### 3.6 Вимоги до транспортування і зберігання

Транспортування повинно проводитися довіреною особою. Воно проводиться в упаковці, яка захищає носії з програмним комплексом від різного роду пошкоджень.

Місце зберігання програмного комплексу повинно бути сухим, з відсутністю пилу та з низьким коефіцієнтом відносної вологості повітря. Строк зберігання програмного комплексу залежить від носія інформації.

Транспортування буде здійснюватися на таких носіях: CD/DVD-RW, флеш, також через глобальну всесвітню систему інформаційного обміну Internet.

#### 4 ВИМОГИ ДО ПРОГРАМНОЇ ДОКУМЕНТАЦІЇ

До складу програмної документації має входити технічне завдання та робочий проект.

До складу робочого проекту мають входити:

- специфікація;
- текст програми;
- опис програми;
- керівництво користувача. Керівництво з моделювання нерегулярної часової послідовності.

Вся документація до програми повинна задовольняти вимогам державного стандарту до оформлення програмних документів [3].

## ВИЗНАЧЕННЯ ВИТРАТ НА ПРОЕКТУВАННЯ ПРОГРАМИ

Техніко-економічне обґрунтування (ТЕО) — це розрахунок економічної доцільності здійснення проекту, заснований на порівняльній оцінці витрат і результатів ефективності використання, а також строку окупності вкладень. ТЕО — це виваженість кожного вашого кроку в реалізації задуманого.

Початковим етапом розрахунку величини трудових витрат розробників є оцінка розміру програмного забезпечення. Основні відмінності методик, що застосовуються в оцінці трудовитрат, полягають у використуваному типі критерію оцінки якості [4].

Згідно моделі COSOMO, розмір проекту  $S$  вимірюється в рядках коду LOC (KLOC), а трудовитрати в людино-місяцях.

$$E = a \cdot S^b \cdot EAF, \quad (5.1)$$

де  $E$  – витрати праці на проект (в людино-місяцях);

$S^b$  – розмір коду (в KLOC);

$EAF$  – фактор уточнення витрат (effort adjustment factor).

Для простих систем,  $a = 2,4$ ;  $b = 1,05$

Припустимо, що розмір програмного коду програмного засобу 899 рядки:

$$E = 2,4 \cdot 0,899^{1,05} \cdot 1 = 2,12$$

Отже, згідно моделі COSOMO, орієнтовні трудовитрати на проект складуть приблизно 2,12 людино-місяці.

Нижче наведені розрахунки вартості розробки «Дослідження методів моделювання часових послідовностей з нерівномірним кроком». Основними статтями витрат прийняті:

- основна заробітна плата;
- відрахування на соціальні потреби;
- накладні витрати;

- витрати на персональний комп’ютер і ліцензійні базові програмні засоби.

Основна заробітна плата (ОЗП) оцінює працю інженера-програміста зі створення програмного продукту і визначається виходячи з кількості розробників, часу виконання розробки (годин), а також заробітної плати в розрахунку на одну годину. Розрахунок заробітної платні проводиться по формі табл. 5.1.

Таблиця 5.1

Фонд місячної заробітної плати

№ п/п	Посада виконавця	Оклад, грн/міс	Кількість		Сума зарплати грн
			чол	місяців	
1	інженер-програміст	17200 [2]	1	2,12	36464

Описаний в проекті програмний продукт був розроблений одним програмістом в період з 19.09.22 до 28.11.22, що складає 70 днів або приблизно 9 робочих тижнів. Витрати робочого часу прийняті за 40 годин у тиждень. Погодинна ставка кваліфікованого інженера–програміста складає 56,25 грн/год. Таким чином, витрачено робочого часу:

$$t_{\text{розробки}} = N_{\text{чол}} \times N_{\text{тиж}} \times N_{\text{год}}, \quad (5.2)$$

де  $N_{\text{чол}}$  – кількість виконавців, чол;

$N_{\text{тиж}}$  – тривалість розробки;

$N_{\text{год}}$  – витрати робочого часу, год;

$$t_{\text{розробки}} = 1 \cdot 10 \cdot 40 = 400 \text{ чол/год.}$$

ОЗП визначається за формулою:

$$OЗП = t_{\text{розробки}} \cdot N \cdot KKB, \quad (5.3)$$

де  $t_{\text{розробки}}$  – витрати праці у чол/год;

$N$  – погодинна ставка;

$KKB$  – коефіцієнт кваліфікації програміста, обумовлений від стажу роботи з даної спеціальності. Коефіцієнт кваліфікації розробника ( $k$ ) - ступінь підготовленості виконавця до дорученої йому роботи (він визначається залежність від стажу праці та

становить:

- для працюючих до 2 років- 0,75;
- від 2 до 3 років 1,0;
- від 3 до 5 років - 1,1-1,2;
- від 5 до 7 років - 1,3-1,4;
- понад 7 років - 1,5-1,6.

В даному випадку  $KKB$  приймається 0,75. ОЗП складає:

$$OЗП = 400 \cdot 107,5 \cdot 0,75 = 32250 \text{ грн.}$$

Відрахування на соціальні потреби встановлюються у відсотках від суми заробітної плати (22% [4]):

$$C_{\text{соц}} = \frac{OЗП \cdot 22\%}{100\%} \quad (5.4)$$

$$C_{\text{соц}} = \frac{32250 \cdot 22\%}{100\%} = 7095 \text{ грн.}$$

Отримані результати за (5.3) та (5.4) підсумовуються. Вони складають 20587,5 грн. та визначають основні прямі витрати.

Накладні витрати враховують загальногосподарчі витрати по забезпеченню проведення роботи: витрати на опалення, електроенергію, амортизація будівель, зарплату адміністративного персоналу та інше. Вони визначаються в процентах (30 – 40%) від суми прямих витрат:

$$\begin{aligned}
& 15 \\
& 44165850.1218-01 \\
C_{\text{накл}} &= \frac{(OЗП+C_{\text{соц}}) \cdot 35\%}{100\%};
\end{aligned}
\tag{5.5}$$

$$C_{\text{накл}} = \frac{(32250+7095) \cdot 35\%}{100\%} = 13770,75 \text{ грн.}$$

Протягом усього терміну використання нової техніки підприємство щорічно витрачає певні кошти, пов'язані з її експлуатацією.

Експлуатаційні витрати на персональний комп'ютер визначаються протягом терміну розробки програмного засобу в залежності від вартості комп'ютеру. В експлуатаційні витрати входять:

- вартість витратних матеріалів;
- витрати на ремонт;
- заробітна плата ремонтника;
- оренда приміщення;
- додаткові витрати – прибирання приміщення, охорона, оренда, комунальні послуги;
- амортизаційні витрати на персональний комп'ютер і програмне забезпечення;
- витрати на електроенергію ( $C_{\text{ел}}$ ),  
які визначаються за формулою:

$$C_{\text{ел}} = P \cdot B \cdot T_{\text{розр}}, \tag{5.6}$$

- де  $P$  – потужність комп'ютера та допоміжних споживачів електричної енергії, приймається 0,62 кВт/год;
- $B$  – вартість 1 кВт/годин для непобутових споживачів,  
– складає 3 грн [3];
- $T_{\text{розр}}$  – час роботи з ЕВМ, приймається рівним робочому часу.

Витрати на електроенергію визначаються так:

$$C_{\text{ел}} = 0,62 \cdot 3 \cdot 400 = 744 \text{ грн.}$$

Витрати на витратні матеріали ( $C_{\text{ВМ}}$ ) протягом всього терміну експлуатації приблизно 10% від вартості комп'ютеру. Вартість робочої станції приймається 15 594 грн. [6,7,8,9,10], термін експлуатації – 4 років.

Найменування	Кількість	Ціна
Материнська плата Gigabyte GA-B75M-D3V	1	1895 грн. [7]
Процесор Intel Core i7-3770	1	3999 грн. [8]
Transcend DIMM TS1GLK72V6H 8GB	2	2801 грн. [9]
Блок живлення Aerocool VX Plus 600	1	1499 грн. [10]
Монітор ASUS VS228NE	1	2599 грн. [6]
Всього		15 594 грн.

Отже, можна визначити ці витрати за період створення програмного засобу:

$$C_{\text{ВМ}} = V_{\text{КОМ}} \cdot \frac{N_{\text{Д}}}{N_{\text{ЕКСП}} \cdot 365} \cdot \frac{10\%}{100\%}, \quad (5.7)$$

де  $V_{\text{КОМ}}$  – вартість персонального комп'ютеру;

$N_{\text{Д}}$  – кількість днів розробки програмного продукту;

$N_{\text{ЕКСП}}$  – термін експлуатації персонального комп'ютеру.

Витрати на витратні матеріали визначаються так:

Витрати на витратні матеріали визначаються так:

$$C_{\text{ВМ}} = 15594 \cdot \frac{70}{4 \cdot 365} \cdot \frac{10}{100} = 74,77 \text{ грн.}$$

Заробітна плата ремонтника ( $C_{\text{рем}}$ ) визначена наступним чином: на ремонт 50 комп'ютерів потрібен один інженер-системотехнік. Його середньомісячна заробітна плата приймається 15000 грн. Тоді в перерахунку на один комп'ютер його заробітна плата за період розробки програмного продукту складає:

$$C_{\text{рем}} = \frac{C'_{\text{рем}}}{N_{\text{КОМ}}} \cdot T_{\text{міс}}, \quad (5.8)$$

де  $C'_{\text{рем}}$  – середньомісячна заробітна плата;

$N_{\text{КОМ}}$  – кількість комп'ютерів на одного ремонтника.

$T_{\text{мес}}$  – час розробки програмного продукту, міс.

Заробітна плата ремонтника ( $C_{\text{рем}}$ ) буде складати:

$$C_{\text{рем}} = \frac{15000}{50} \cdot 2,12 = 636 \text{ грн.}$$

За статистикою витрати на комплектуючі вироби ( $C_{\text{КОМ}}$ ) для ремонту персонального комп'ютера складає 10% від його вартості за термін його експлуатації, тобто рівні витратам на витратні матеріали:

$$C_{КОМ} = C_{ВМ} = 74,77 \text{ грн.} \quad (5.9)$$

Амортизаційні відрахування на персональний комп'ютер (АПК) визначені з положення, що амортизаційний період в даний час дорівнює терміну морального старіння обчислювальної техніки і складає 4 роки. Отже, за 4 роки амортизаційні відрахування на персональний комп'ютер дорівнюють вартості комп'ютера. За період проектування амортизаційні відрахування складуть:

$$АКП = В_{КОМ} \cdot \frac{N_{Д}}{N_{експ} \cdot 365}; \quad (5.10)$$

$$АКП = 15594 \cdot \frac{2,12}{4 \cdot 12} = 688,74 \text{ грн.}$$

Амортизаційні відрахування на програмне забезпечення (АПЗ) залежать від його циклу заміни. Якщо прийняти термін морального старіння для Windows 2 років та Visual Studio за 1 рік то амортизаційні відрахування на програмне забезпечення дорівнюють його вартості.

Для функціонування персонального комп'ютера використовувалася операційна система Windows 10, для написання програмного забезпечення - програмне середовище Visual Studio 2022 Professional.

$$АПЗ_w = 7899 \cdot \frac{2,12}{2 \cdot 12} = 697,75 \text{ грн}$$

$$АПЗ_w = 1850 \cdot \frac{2,12}{1 \cdot 12} = 326,83 \text{ грн}$$

Розрахунок амортизаційних відрахувань на програмне забезпечення зведений в табл. 5.2. Додаткові витрати ( $C_{дод}$ ): прибирання приміщень, охорона, комунальні послуги важко оцінити точно і прийняти рівними 50% заробітної плати інженера-програміста, тобто 8600 гривень на місяць. Тобто за весь період розробки – 18232 грн.

Оренду приміщень для однієї людини приймемо рівною 4500 гривень на місяць [11]. Тобто за весь період розробки – 11250 грн. Сумарні експлуатаційні витрати на один персональний комп'ютер складають:

$$C_{експ} = C_{ел} + C_{ВМ} + C_{рем} + АКП + АПЗ + C_{ор} + C_{дод}; \quad (5.11)$$

$$\text{Сексп} = 744 + 74,77 + 636 + 688,74 + 1024,58 + 11250 + 18232 = 32650,09 \text{ грн}$$

Результати розрахунків зведено у табл. 5.3.

Таблиця 5.2

## Використовуване програмне забезпечення

Найменування програмного забезпечення	Вартість програмного забезпечення, грн	Джерело придбання	Амортизаційні відрахування, грн
Windows 10	7899	<a href="https://www.microsoft.com/uk-ua/d/windows-10-pro/df77x4d43rkt?rtc=1&amp;activetab=pivot:оглядtab&amp;source=lp">https://www.microsoft.com/uk-ua/d/windows-10-pro/df77x4d43rkt?rtc=1&amp;activetab=pivot:оглядtab&amp;source=lp</a>	697,75
Visual Studio 2022 Professional	1850	<a href="https://likesoft.com.ua/ru/ofisnoe-po/1056/297/?gclid=Cj0KCQiAkMGcBhCSARIsAIW6d0D4tOc7nRDoYXdO_RngP7BiyZ9lFI4EJP0tdil7MhTc4AT9ecJ78aAm2nEALw_wcB">https://likesoft.com.ua/ru/ofisnoe-po/1056/297/?gclid=Cj0KCQiAkMGcBhCSARIsAIW6d0D4tOc7nRDoYXdO_RngP7BiyZ9lFI4EJP0tdil7MhTc4AT9ecJ78aAm2nEALw_wcB</a>	326,83
Всього:	9749		1024,58

Таблиця 5.3

## Експлуатаційні витрати на ПК і ПЗ.

Найменування витрат	Витрати, грн
---------------------	--------------

Витрати на електроенергію	744
Вартість витратних матеріалів	74,77
Витрати на ремонт	636
Амортизація персонального комп'ютера	1252,25
Амортизація програмного забезпечення	688,74
Оренда приміщення	11250
Додаткові витрати	18232
Всього	32650,09

Таким чином, витрати на створення програмного продукту складають:

$$C_{\text{розробки}} = OЗП + C_{\text{соц}} + C_{\text{накл}} + C_{\text{експ}}; \quad (5.12)$$

$$C_{\text{розробки}} = 3646 + 7095 + 13770,75 + 32650,09 = 89979 \text{ грн.}$$

Розрахунок витрат зведено у табл. 5.4.

Таблиця 5.4

#### Кошторис витрат на розробку програмного засобу

Найменування витрат	Витрати, грн
Основна заробітна плата	36464
Відрахування на соціальні потреби	7095
Накладні витрати	13770,75
Експлуатаційні витрати	32650,09
Всього	89979

За отриманими значеннями техніко-економічних показників проекту складено кошторис витрат на розробку сучасного програмного забезпечення для оцінки схожості програм. За результатами розрахунків, приблизна вартість розробки складає 89979 грн.

Список матеріалів

1. Формули 5.1 – 5.12  
<http://repository.enu.kz/bitstream/handle/data/12881/METODIKA-TRUDOZATRAT.pdf>
2. <https://www.work.ua/salary-инженер-программист/>
3. <https://www.kmu.gov.ua/npas/pro-vnesennya-zmin-do-postanovi-ka-a859>
4. <https://services.dtkr.ua/catalogues/indexes/13>
5. <https://index.minfin.com.ua/ua/tariff/electric/prom/>
6. [https://www.moyo.ua/ua/monitor\\_asus\\_21\\_5\\_vs228ne\\_90lmd8501t02211c-/301260.html](https://www.moyo.ua/ua/monitor_asus_21_5_vs228ne_90lmd8501t02211c-/301260.html)
7. <https://hard.rozetka.com.ua/ua/254668621/p254668621/>
8. <https://hard.rozetka.com.ua/ua/116031487/p116031487/>
9. <https://hard.rozetka.com.ua/ua/226991875/p226991875/>
10. [https://hard.rozetka.com.ua/ua/aerocool\\_vx\\_plus\\_600\\_600w/p107726364/](https://hard.rozetka.com.ua/ua/aerocool_vx_plus_600_600w/p107726364/)
11. [https://www.olx.ua/d/nedvizhimost/kommercheskaya-nedvizhimost/arenda-kommercheskoy-nedvizhimosti/dnepr/?currency=UAH&search%5Bfilter\\_float\\_total\\_area:from%5D=14&search%5Bfilter\\_float\\_total\\_area:to%5D=20&view=list](https://www.olx.ua/d/nedvizhimost/kommercheskaya-nedvizhimost/arenda-kommercheskoy-nedvizhimosti/dnepr/?currency=UAH&search%5Bfilter_float_total_area:from%5D=14&search%5Bfilter_float_total_area:to%5D=20&view=list)

21  
44165850.1218-01  
**6 СТАДІЇ ТА ЕТАПИ РОЗРОБКИ**

Усі стадії та етапи розробки приведені у табл.6.1.

Таблиця 6.1

Етапи розробки та строки

Стаді розробки	Етап розробки	Термін
Технічне завдання (ТЗ)	Постановка задачі, збір початкових матеріалів	15.11.20 – 02.12.20
	Розробка структур вхідних та вихідних даних	02.12.20 – 12.01.21
	Визначення вимог до програми	12.01.20 – 25.01.21
	Узгодження та затвердження технічного завдання	25.01.21 – 19.02.21
Робочий проект	Програмування та відладка програми	19.02.21 – 10.07.21
	Тестування програми	10.07.21 – 01.09.21
Впровадження	Розробка, узгодження та затвердження програмної документації	01.09.21 – 20.12.21

## 7 ПОРЯДОК КОНТРОЛЮ ТА ПРИЙОМУ

Контроль здійснюється за допомогою виконання набору тестів з метою знаходження помилок в програмі та його специфікації. Контроль виконання роботи забезпечується головним керівником розробки.

Прийом програми здійснюється уповноваженою комісією.

## СПИСОК ЛІТЕРАТУРИ

1. ДСанПіН 3.3.2-007-98. Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин [Текст] / Постанова Головного державного санітарного лікаря України від 10 грудня 1998 р. № 7 – К., 1998.
2. Основи стандартизації програмних систем [Текст]: методичні вказівки до дипломного проектування та лабораторних робіт / уклад.: Ю. М. Івченко, В. І. Шинкаренко, В. Г. Івченко; Дніпропетр. нац. ун-т залізн. трансп. ім. акад. В. Лазаряна. – Д.: Вид-во Дніпропетр. нац. ун-ту залізн. трансп. ім. акад. В. Лазаряна, 2009. – 38 с.
3. Медведкова І. В. Порівняльний аналіз методів оцінки вартості проектів з розробки програмного забезпечення [Електронний ресурс] /І. В. Медведкова, А. А. Іванов [Москва: національний дослідницький університет МІФІ] – №7. – Режим доступу cyberleninka.ru (дата звернення 15.04.2019 р.).
4. Інженер-програміст: середня зарплата в Україні [Електронний ресурс] – Режим доступу:<http://www.wikipage.com.ua/1x4306.html>tech.informator.ua
5. Тарифи на електроенергію в 2019 році [Електронний ресурс] – Режим доступу: [www.prostobank.ua](http://www.prostobank.ua)
6. Комп'ютерник: середня зарплата в Україні [Електронний ресурс] – Режим доступу: <https://www.work.ua>
7. Статистика цін на оренду квартир в Дніпропетровську [Електронний ресурс] – Режим доступу: [100realty.ua](http://100realty.ua)
8. Бутстреп, малі вибірки, застосування в аналізі даних [Електронний ресурс] – Режим доступу: <http://statistica.ru/theory/metod-butstrepa-i-ego-primenenie-v-sovremennom-analize-dannykh/>

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ЗАТВЕРДЖЕНО  
1116130. 01165-01 ІЗ 01 ЛЗ

«Дослідження методів моделювання часових  
послідовностей з нерівномірним кроком»

Керівництво користувача. Керівництво з моделювання нечіткої часової  
послідовності

1116130. 01165-01 ІЗ 01

Аркушів 12

## АНОТАЦІЯ

Документ 1116130.01165-01 ІЗ «Дослідження методів моделювання часових послідовностей з нерівномірним кроком» Керівництво користувача. Керівництво з моделювання.

У даному документі представлено призначення та умови застосування програми, підготовка до роботи, опис операцій, аварійні ситуації, рекомендації щодо засвоєння.

ЗМІСТ

Вступ.....	27
1. Призначення та умови застосування.....	28
2. Підготовка до роботи.....	29
3. Опис операцій.....	30
4. Аварійні ситуації.....	34

## ВСТУП

Програмний комплекс надає можливість користувачеві обробляти процеси, будь якого виду, від технологічних процесів виробництва до динаміки медичних обстежень та інших складних систем. Забезпечуючи широкий спектр функціональності, програма дозволяє використовувати різноманітні методи та алгоритми для аналізу та моделювання нерівномірних часових послідовностей. Для користування програмним комплексом користувач повинен володіти базовими знаннями користування персональним комп'ютером. Для того щоб користувач міг використовувати програмний комплекс у повному обсязі, необхідно ознайомитися з програмною документацією опис програми та керівництво користувача.

## 1. ПРИЗНАЧЕННЯ ТА УМОВИ ЗАСТОСУВАННЯ

Програмний комплекс призначений для дослідження методів моделювання часових послідовностей з нерівномірним кроком. Це дозволяє зробити роботу лікаря більш продуктивною та зекономити його робочий час, а також уникнути критичних ситуацій у стані хворого.

Умови для використання програмного комплексу такі:

- ЕОМ з встановленою операційною системою Windows;
- початковий набір даних відвідувань
- монітор
- клавіатура
- миша

Мінімальні вимоги до характеристик ЕОМ для комфортного використання такі:

- 512 МБ пам'яті на жорсткому диску
- 2 ГБ ОЗУ
- Процесор на 2 ядра з тактовою частотою 3.0 GHz
- USB-роз'єм.

## 2. ПІДГОТОВКА ДО РОБОТИ

Для виклику програмного комплексу необхідно запустити файл під назвою Forecast.exe. Для перевірки програмного комплексу необхідно підготувати початкові дані, налаштувати та запустити алгоритм прогнозування. Дані перевірки програми готуються у вигляді файлу з послідовністю даних про відвідування.

### 3. ОПИС ОПЕРАЦІЙ

Після запуску програмного комплексу, з'явиться головне вікно програми в якому виконуються основні функції, описані нижче:

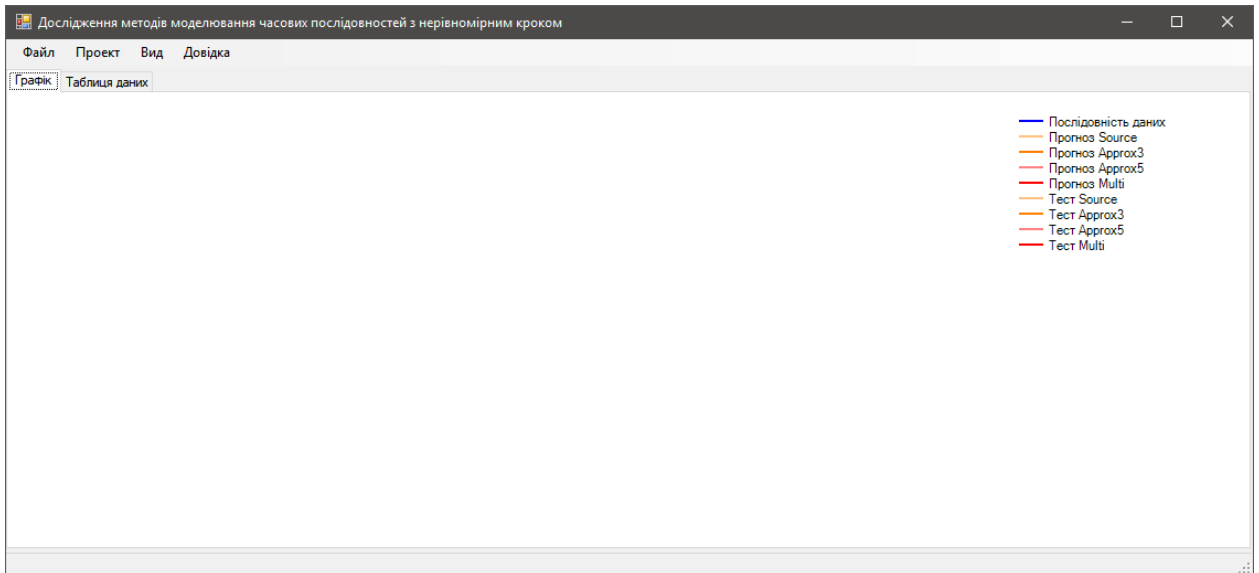


Рисунок 3.1 – Головне вікно програми

Для початку роботи з програмою, користувачу потрібно завантажити початкові дані для отримання прогнозу на їх основі. Зробити це можливо за допомогою пункту меню файл-> Створити або Відкрити, якщо проект вже існує. Пункт меню зображений на малюнку 3.2.

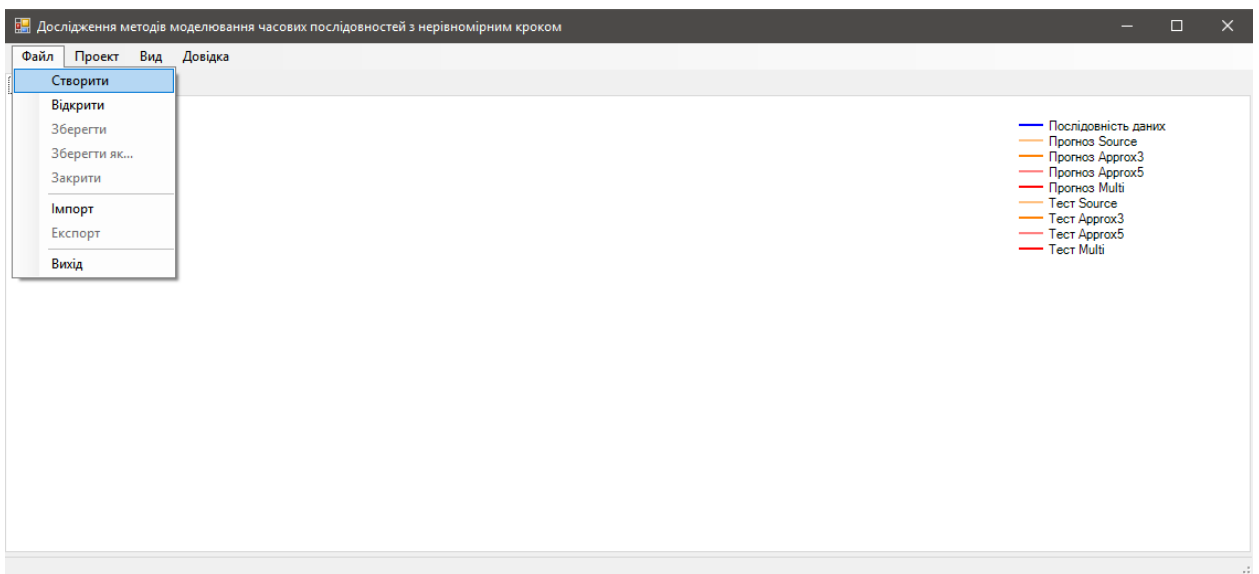


Рисунок 3.2 – Пункти меню «Файл»

В цьому ж меню програми користувач може зберегти проект, імпортувати або експортувати дані в формати інших програмних засобів та завершити роботу. Без збереження даних отриманий прогноз та дані будуть втрачені. Ці пункти меню також можна побачити на рисунку 3.2.

Якщо користувач бажає завершити роботу та вийти з програми передбачено пункт меню «Вихід».

Створив новий проект з даними, або відкривши існуючий, користувач відразу побачить дані у вигляді графіку на одноіменній сторінці головного вікна програми, який зображено на рисунку 3.3.

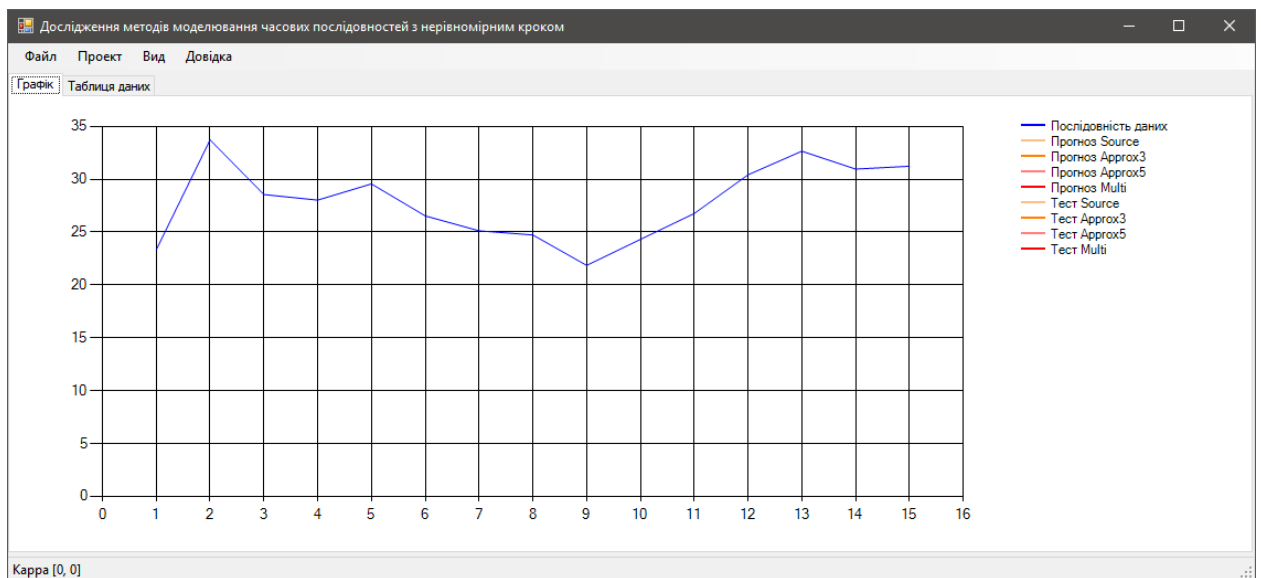
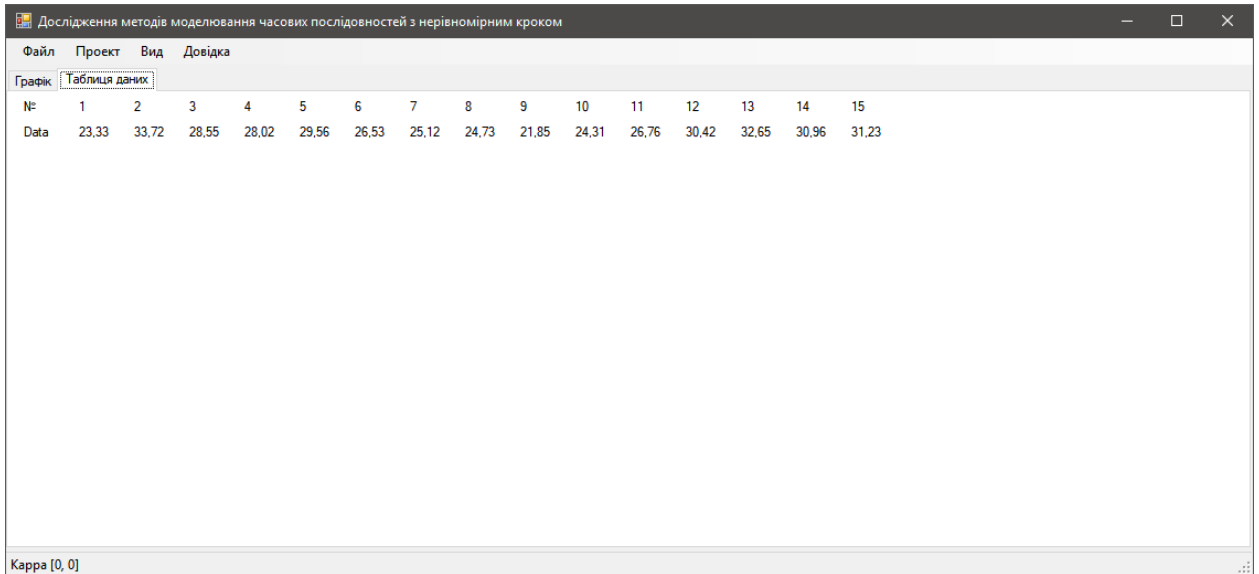


Рисунок 3.3 – Дані у вигляді графіку

Також всі дані будуть відображені на другій сторінці «Таблиця» у вигляді таблиці, що можна побачити на рисунку 3.4



Дослідження методів моделювання часових послідовностей з нерівномірним кроком

Файл Проект Вид Довідка

Графік Таблиця даних

№	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Data	23,33	33,72	28,55	28,02	29,56	26,53	25,12	24,73	21,85	24,31	26,76	30,42	32,65	30,96	31,23

Карра [0, 0]

Рисунок 3.4 – Дані у вигляді таблиці

Наступний пункт меню називається «Проект». Він містить налаштування алгоритму прогнозування для отримання більш точних результатів. Розглянути його можна на рисунку 3.5

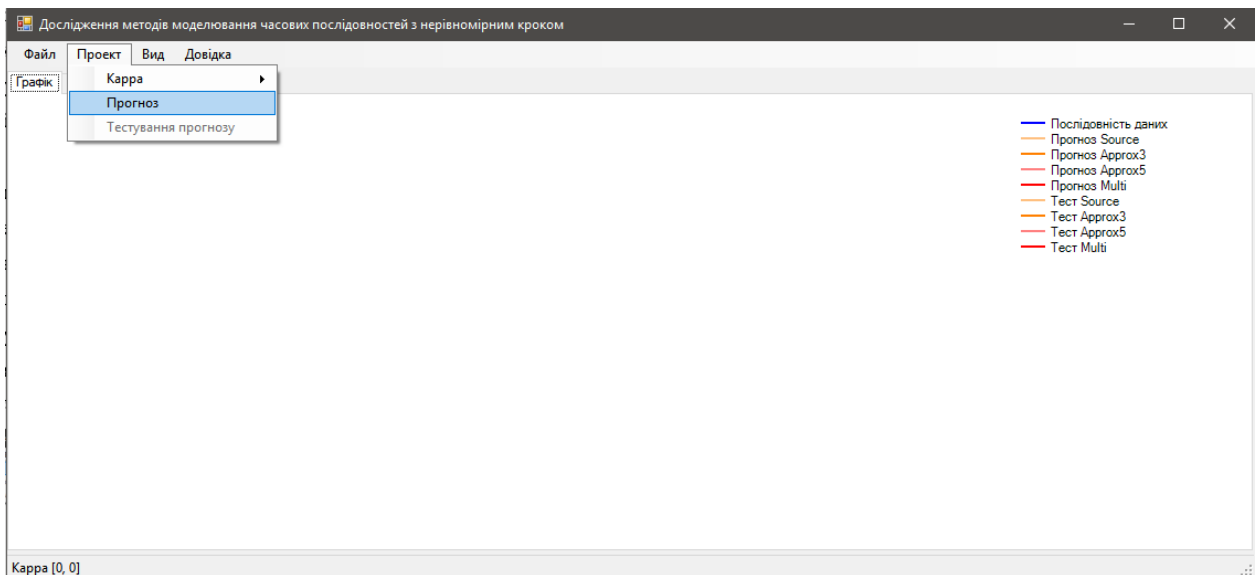


Рисунок 3.5 – Пункти меню «Проект»

Після того як користувач створить новий проект, йому потрібно задати значення Карра, вибрав одноіменний пункт в меню «Проект». Користувач побачить вікно для ведення значення та діапазону у відсотках. Розглянути це вікно можна на рисунку 3.6.

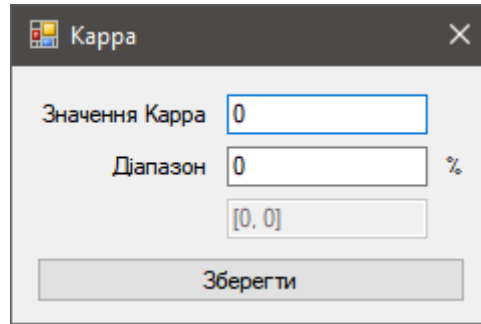


Рисунок 3.6 – Вікно встановлення значення Карра

Після цього користувач може обрати пункт меню «Прогноз», та обрати у з'явившомуся вікні кількість кроків прогнозування. Це вікно можна побачити на рисунку 3.7.

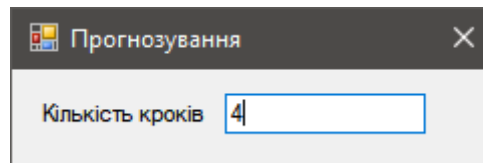


Рисунок 3.7 – Вікно встановлення кроків прогнозування

Після цієї дії програма відразу зробить прогноз на встановлену кількість кроків, починаючи з кінця заданої числової послідовності. Оновлені дані можна буде побачити на обох сторінках головного вікна, що можна побачити на рисунку 3.8 та рисунку 3.9.

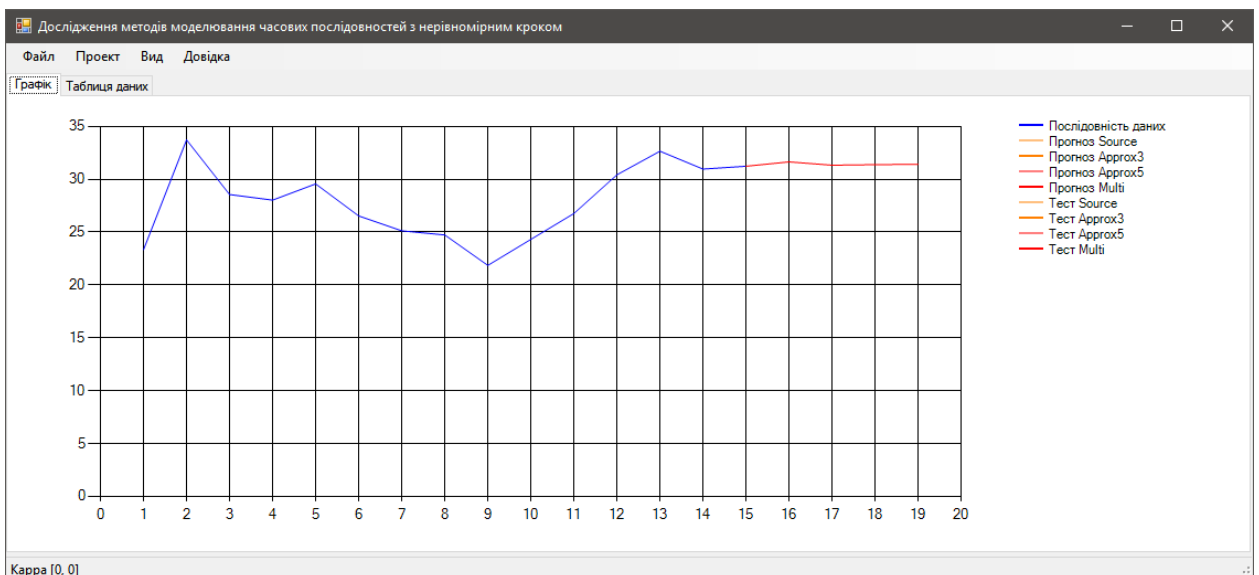


Рисунок 3.8 – Доповнений після прогнозування графік

№	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
Data	23.33	33.72	28.55	28.02	29.56	26.53	25.12	24.73	21.85	24.31	26.76	30.42	32.65	30.96	31.23				
Multi																31.64	31.33	31.38	31.41

Рисунок 3.9 – Доповнена після прогнозування таблиця

При необхідності користувач може приховати непотрібні послідовності з графіку та таблиці використовуючи меню «Вид», де натисканням на окремі пункту або групи можна перемикати їх видимість. Це меню можна побачити на рисунку 3.10.

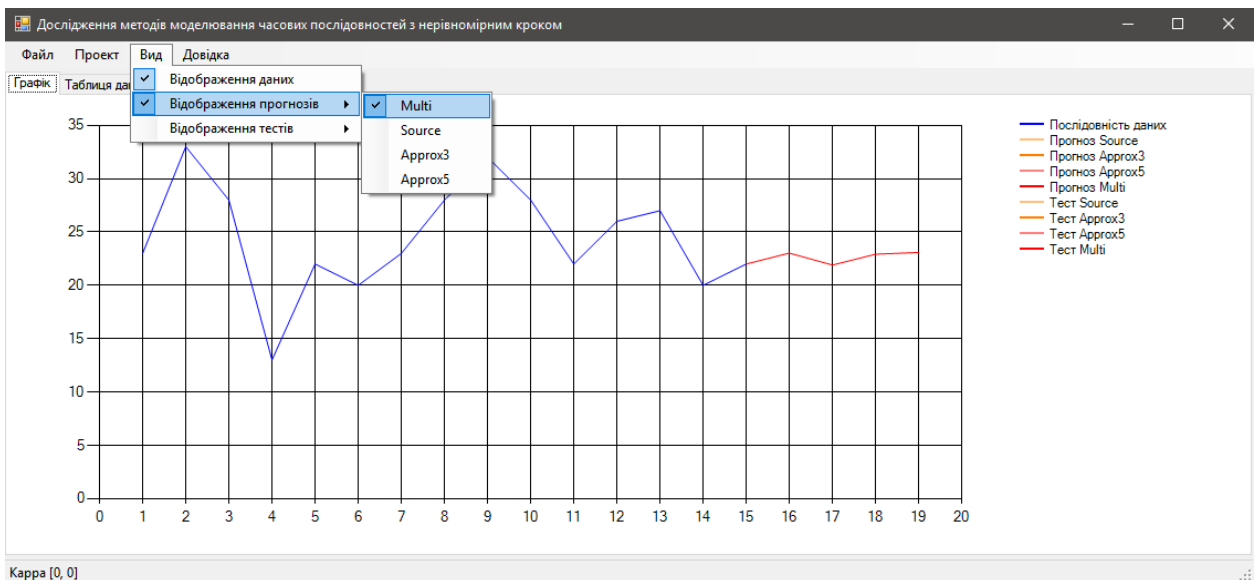


Рисунок 3.10 – Пункти меню «Вид»

Також користувач може перевірити точність алгоритму, зробив паралельний прогноз з існуючими даними. Для цього він може натиснути у меню «Проект» кнопку «Тестування прогнозу», після чого ці послідовності

будуть відображені на сторінках «Графік» та «Таблиця даних». Графік з тестом алгоритму можна побачити на рисунку 3.11.

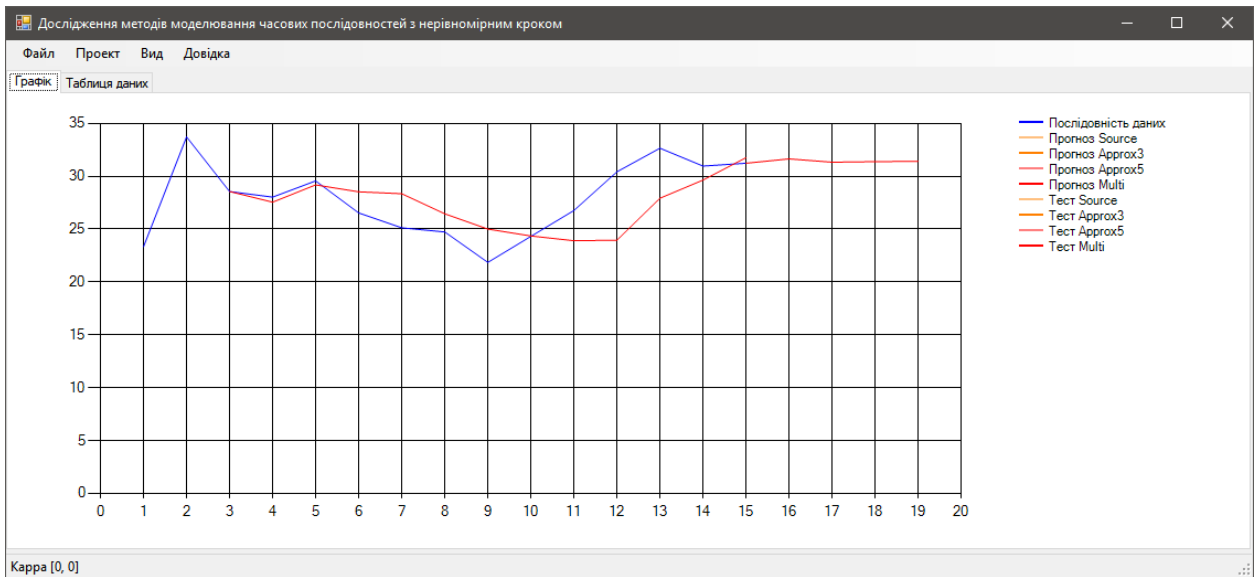


Рисунок 3.11 – Графік при тестуванні

Використовуючи зазначене вище меню «Вид» також можна приховати, або відобразити проміжні дані тестового алгоритму, використовуючи підменю «Відображення тестів»

#### 4. АВАРІЙНІ СИТУАЦІЇ

При недостатній кількості вхідних даних програма може викинути критичну помилку та не здатна буде здійснити прогноз, оскільки прогноз може бути побудований лише на основі початково наданих даних.

Якщо програма завершиться без збереження отриманих під час прогнозу даних, вони будуть втрачені.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ЗАТВЕРДЖЕНО  
44165850.1218-01 12 01-ЛЗ

«Дослідження методів моделювання часових  
послідовностей з нерівномірним кроком»

Текст програми

1116130. 01165-01 12 01

Аркушів 25

## АНОТАЦІЯ

Документ 1116130.01165-01 12 «Дослідження методів моделювання часових послідовностей з нерівномірним кроком». Текст програми, що входить до складу програмної документації до дипломного проекту.

У даному документі представлена структура програми, текст програми.

### Текст програми 1 – ForecastForm.cpp

```
#include "ForecastForm.h"
#include "KappaForm.h"
#include "ForecastStepForm.h"

using namespace System;
using namespace System::Windows::Forms;

double myRound(double _value, int _after = 0);

[STAThreadAttribute]
int main(/*array<String> args*/) {
    Application::SetCompatibleTextRenderingDefault(false);
    Application::EnableVisualStyles();
    Forecast::ForecastForm form;
    Application::Run(% form);

    return 0;
}

ForecastDS TestForecast;

System::Void
Forecast::ForecastForm::form_button_test_Click(System::Object ^sender,
System::EventArgs ^e) {
    //MessageBox.Show("О программе", MessageBoxButtons::YesNo,
    MessageBoxIcon::Question);
    MessageBox::Show("Выполнено", "Окно", MessageBoxButtons::OKCancel);

    return System::Void();
}
```

```

System::Void
Forecast::ForecastForm::form_ToolStrip_File_Exit_Click(System::Object
^sender, System::EventArgs ^e) {
    if (MessageBox::Show("Ви впевнені?", "Вихід",
MessageBoxButtons::YesNo, MessageBoxIcon::Warning) ==
System::Windows::Forms::DialogResult::Yes) {
        Application::Exit();
    }

    UpdateGraph();
    UpdateTable();

    return System::Void();
}

```

```

System::Void
Forecast::ForecastForm::form_menustrip_view_data_Click(System::Object
^sender, System::EventArgs ^e) {
    if (this->form_menustrip_view_data->Checked == true) {
        this->form_menustrip_view_data->Checked = false;
    }
    else {
        this->form_menustrip_view_data->Checked = true;
    }

    UpdateGraph();
    UpdateTable();

    return System::Void();
}

```

```

System::Void
Forecast::ForecastForm::form_menustrip_view_forecast_Click(System::Object
^sender, System::EventArgs ^e) {
    if (this->form_menustrip_view_forecast->Checked == true) {
        this->form_menustrip_view_forecast->Checked = false;
    }
    else {
        this->form_menustrip_view_forecast->Checked = true;
    }

    UpdateGraph();
}

```

```
UpdateTable();
```

```
return System::Void();
```

```
}
```

```
System::Void
```

```
Forecast::ForecastForm::form_menustrip_view_forecast_source_Click(System::Object ^sender, System::EventArgs ^e) {
```

```
    if (this->form_menustrip_view_forecast_source->Checked == true) {
```

```
        this->form_menustrip_view_forecast_source->Checked = false;
```

```
    }
```

```
    else {
```

```
        this->form_menustrip_view_forecast_source->Checked = true;
```

```
    }
```

```
UpdateGraph();
```

```
UpdateTable();
```

```
return System::Void();
```

```
}
```

```
System::Void
```

```
Forecast::ForecastForm::form_menustrip_view_forecast_approx5_Click(System::Object ^sender, System::EventArgs ^e) {
```

```
    if (this->form_menustrip_view_forecast_approx5->Checked == true) {
```

```
        this->form_menustrip_view_forecast_approx5->Checked = false;
```

```
    }
```

```
    else {
```

```
        this->form_menustrip_view_forecast_approx5->Checked = true;
```

```
    }
```

```
UpdateGraph();
```

```
UpdateTable();
```

```
return System::Void();
```

```
}
```

```
System::Void
```

```
Forecast::ForecastForm::form_menustrip_view_forecast_multi_Click(System::Object ^sender, System::EventArgs ^e) {
```

```
    if (this->form_menustrip_view_forecast_multi->Checked == true) {
```

```
        this->form_menustrip_view_forecast_multi->Checked = false;
```

```

    }
    else {
        this->form_menustrip_view_forecast_multi->Checked = true;
    }

    UpdateGraph();
    UpdateTable();

    return System::Void();
}

System::Void
Forecast::ForecastForm::form_menustrip_view_forecastttest_source_Click(System
::Object ^sender, System::EventArgs ^e) {
    if (this->form_menustrip_view_forecastttest_source->Checked == true)
    {
        this->form_menustrip_view_forecastttest_source->Checked = false;
    }
    else {
        this->form_menustrip_view_forecastttest_source->Checked = true;
    }

    UpdateGraph();
    UpdateTable();

    return System::Void();
}

System::Void
Forecast::ForecastForm::form_menustrip_view_forecastttest_approx3_Click(Syste
m::Object ^sender, System::EventArgs ^e) {
    if (this->form_menustrip_view_forecastttest_approx3->Checked ==
true) {
        this->form_menustrip_view_forecastttest_approx3->Checked =
false;
    }
    else {
        this->form_menustrip_view_forecastttest_approx3->Checked = true;
    }

    UpdateGraph();
    UpdateTable();
}

```

```

        return System::Void();
    }

    System::Void
Forecast::ForecastForm::form_menustrip_view_forecastttest_approx5_Click(System
m::Object ^sender, System::EventArgs ^e) {
    if (this->form_menustrip_view_forecastttest_approx5->Checked ==
true) {
        this->form_menustrip_view_forecastttest_approx5->Checked =
false;
    }
    else {
        this->form_menustrip_view_forecastttest_approx5->Checked = true;
    }

    UpdateGraph();
    UpdateTable();

    return System::Void();
}

    System::Void
Forecast::ForecastForm::form_menustrip_view_forecastttest_multi_Click(System:
:Object ^sender, System::EventArgs ^e) {
    if (this->form_menustrip_view_forecastttest_multi->Checked == true)
{
        this->form_menustrip_view_forecastttest_multi->Checked = false;
    }
    else {
        this->form_menustrip_view_forecastttest_multi->Checked = true;
    }

    UpdateGraph();
    UpdateTable();

    return System::Void();
}

    System::Void
Forecast::ForecastForm::form_menustrip_view_forecast_approx3_Click(System::O
bject ^sender, System::EventArgs ^e) {

```

```

if (this->form_menustrip_view_forecast_approx3->Checked == true) {
    this->form_menustrip_view_forecast_approx3->Checked = false;
}
else {
    this->form_menustrip_view_forecast_approx3->Checked = true;
}

UpdateGraph();
UpdateTable();

return System::Void();
}

System::Void
Forecast::ForecastForm::form_menustrip_view_forecasttest_Click(System::Objec
t ^sender, System::EventArgs ^e) {
    if (this->form_menustrip_view_forecasttest->Checked == true) {
        this->form_menustrip_view_forecasttest->Checked = false;
    }
    else {
        this->form_menustrip_view_forecasttest->Checked = true;
    }

    UpdateGraph();
    UpdateTable();

    return System::Void();
}

void Forecast::ForecastForm::UpdateGraph() {
    vector<double> _buff, _buff_fs, _buff_fa3, _buff_fa5, _buff_fm,
_buff_tfs, _buff_tfa3, _buff_tfa5, _buff_tfm;

    this->form_graph->Series[0]->Points->Clear();
    _buff = TestForecast.getDatasequence();
    if (this->form_menustrip_view_data->Checked == true) {
        for (int i = 0; i < _buff.size(); i++) {
            this->form_graph->Series[0]->Points->AddXY((i + 1),
_buff[i]);
        }
    }
}

```

44165850.1218-01 12 01

```

this->form_graph->Series[1]->Points->Clear();
this->form_graph->Series[2]->Points->Clear();
this->form_graph->Series[3]->Points->Clear();
this->form_graph->Series[4]->Points->Clear();

if (this->form_menustrip_view_forecast->Checked == true) {
    _buff_fs = TestForecast.getForecastSource();
    _buff_fa3 = TestForecast.getForecastApprox3();
    _buff_fa5 = TestForecast.getForecastApprox5();
    _buff_fm = TestForecast.getForecastMulti();

    if (this->form_menustrip_view_data->Checked == true) {
        if (this->form_menustrip_view_forecast_source->Checked ==
true) {
            this->form_graph->Series[1]->Points-
>AddXY(_buff.size(), _buff[_buff.size() - 1]);
        }
        if (this->form_menustrip_view_forecast_approx3->Checked ==
true) {
            this->form_graph->Series[2]->Points-
>AddXY(_buff.size(), _buff[_buff.size() - 1]);
        }
        if (this->form_menustrip_view_forecast_approx5->Checked ==
true) {
            this->form_graph->Series[3]->Points-
>AddXY(_buff.size(), _buff[_buff.size() - 1]);
        }
        if (this->form_menustrip_view_forecast_multi->Checked ==
true) {
            this->form_graph->Series[4]->Points-
>AddXY(_buff.size(), _buff[_buff.size() - 1]);
        }
    }

    for (int i = 0; i < _buff_fm.size(); i++) {
        if (this->form_menustrip_view_forecast_source->Checked ==
true) {
            this->form_graph->Series[1]->Points->AddXY((i + 1) +
_buff.size(), _buff_fs[i]);
        }
        if (this->form_menustrip_view_forecast_approx3->Checked ==
true) {

```

```

        this->form_graph->Series[2]->Points->AddXY((i + 1) +
_buff.size(), _buff_fa3[i]);
    }
    if (this->form_menustrip_view_forecast_approx5->Checked ==
true) {
        this->form_graph->Series[3]->Points->AddXY((i + 1) +
_buff.size(), _buff_fa5[i]);
    }
    if (this->form_menustrip_view_forecast_multi->Checked ==
true) {
        this->form_graph->Series[4]->Points->AddXY((i + 1) +
_buff.size(), _buff_fm[i]);
    }
}

this->form_graph->Series[5]->Points->Clear();
this->form_graph->Series[6]->Points->Clear();
this->form_graph->Series[7]->Points->Clear();
this->form_graph->Series[8]->Points->Clear();

if (this->form_menustrip_view_forecastttest->Checked == true) {
    _buff_tfs = TestForecast.getTestForecastSource();
    _buff_tfa3 = TestForecast.getTestForecastApprox3();
    _buff_tfa5 = TestForecast.getTestForecastApprox5();
    _buff_tfm = TestForecast.getTestForecastMulti();

    if (this->form_menustrip_view_forecastttest_source->Checked ==
true) {
        this->form_graph->Series[5]->Points->AddXY(3, _buff[2]);
    }
    if (this->form_menustrip_view_forecastttest_approx3->Checked ==
true) {
        this->form_graph->Series[6]->Points->AddXY(3, _buff[2]);
    }
    if (this->form_menustrip_view_forecastttest_approx5->Checked ==
true) {
        this->form_graph->Series[7]->Points->AddXY(3, _buff[2]);
    }
    if (this->form_menustrip_view_forecastttest_multi->Checked ==
true) {
        this->form_graph->Series[8]->Points->AddXY(3, _buff[2]);
    }
}

```



```

_buff = TestForecast.getDatasequence();
_buff_fm = TestForecast.getForecastMulti();
_buff_fs = TestForecast.getForecastSource();
_buff_fa3 = TestForecast.getForecastApprox3();
_buff_fa5 = TestForecast.getForecastApprox5();

_buff_tfs = TestForecast.getTestForecastSource();
_buff_tfa3 = TestForecast.getTestForecastApprox3();
_buff_tfa5 = TestForecast.getTestForecastApprox5();
_buff_tfm = TestForecast.getTestForecastMulti();

form_table->Controls->Clear();
form_table->RowStyles->Clear();
form_table->ColumnStyles->Clear();

form_table->RowCount = 1;
form_table->ColumnCount = 1;

if (this->form_menustrip_view_data->Checked == true || this-
>form_menustrip_view_forecast->Checked == true || this-
>form_menustrip_view_forecasttest->Checked == true) {
    //form_table->RowCount++;
    //form_table->RowStyles->Add(gcnew
RowStyle(SizeType::AutoSize));
    label_0->Text = "№";
    this->form_table->Controls->Add(label_0, 0, 0);
}
else {
    //form_table->RowCount++;
    //form_table->RowStyles->Add(gcnew
RowStyle(SizeType::AutoSize));
    label_0->Text = "";
    this->form_table->Controls->Add(label_0, 0, 0);
}
//Определение количества столбцов
int final = 0, start1 = 1, start2 = 1;
if (this->form_menustrip_view_data->Checked == true) {
    final += _buff.size();
    start2 = 3;
}
else {
    if (this->form_menustrip_view_forecasttest->Checked == true) {

```

```

        final += _buff_tfm.size();
    }
}
if (this->form_menustrip_view_forecast->Checked == true) {
    start1 = final;
    final += _buff_fm.size();
}
//Описание строк
if (this->form_menustrip_view_data->Checked == true) {
    form_table->RowCount++;
    form_table->RowStyles->Add(gcnew
RowStyle(SizeType::AutoSize));
    _d = form_table->RowCount - 1;

    label_1->Text = "Data";
    this->form_table->Controls->Add(label_1, 0, _d);
}
if (this->form_menustrip_view_forecast->Checked == true) {
    if (this->form_menustrip_view_forecast_source->Checked == true)
{
        form_table->RowCount++;
        form_table->RowStyles->Add(gcnew
RowStyle(SizeType::AutoSize));
        _fs = form_table->RowCount - 1;

        label_2->Text = "Source";
        this->form_table->Controls->Add(label_2, 0, _fs);
    }
    if (this->form_menustrip_view_forecast_approx3->Checked ==
true) {
        form_table->RowCount++;
        form_table->RowStyles->Add(gcnew
RowStyle(SizeType::AutoSize));
        _fa3 = form_table->RowCount - 1;

        label_3->Text = "Approx3";
        this->form_table->Controls->Add(label_3, 0, _fa3);
    }
    if (this->form_menustrip_view_forecast_approx5->Checked ==
true) {
        form_table->RowCount++;

```

```
form_table->RowStyles->Add(gcnew
RowStyle(SizeType::AutoSize));
    _fa5 = form_table->RowCount - 1;

    label_4->Text = "Approx5";
    this->form_table->Controls->Add(label_4, 0, _fa5);
}
if (this->form_menustrip_view_forecast_multi->Checked == true)
{
    form_table->RowCount++;
    form_table->RowStyles->Add(gcnew
RowStyle(SizeType::AutoSize));
    _fm = form_table->RowCount - 1;

    label_5->Text = "Multi";
    this->form_table->Controls->Add(label_5, 0, _fm);
}
}
if (this->form_menustrip_view_forecastttest->Checked == true) {
    if (this->form_menustrip_view_forecastttest_source->Checked ==
true) {
        form_table->RowCount++;
        form_table->RowStyles->Add(gcnew
RowStyle(SizeType::AutoSize));
        _tfs = form_table->RowCount - 1;

        label_6->Text = "TestSource";
        this->form_table->Controls->Add(label_6, 0, _tfs);
    }
    if (this->form_menustrip_view_forecastttest_approx3->Checked ==
true) {
        form_table->RowCount++;
        form_table->RowStyles->Add(gcnew
RowStyle(SizeType::AutoSize));
        _tfa3 = form_table->RowCount - 1;

        label_7->Text = "TestApprox3";
        this->form_table->Controls->Add(label_7, 0, _tfa3);
    }
    if (this->form_menustrip_view_forecastttest_approx5->Checked ==
true) {
        form_table->RowCount++;
```

```

        form_table->RowStyles->Add(gcnew
RowStyle(SizeType::AutoSize));
        _tfa5 = form_table->RowCount - 1;

        label_8->Text = "TestApprox5";
        this->form_table->Controls->Add(label_8, 0, _tfa5);
    }
    if (this->form_menustrip_view_forecasttest_multi ->Checked ==
true) {
        form_table->RowCount++;
        form_table->RowStyles->Add(gcnew
RowStyle(SizeType::AutoSize));
        _tfm = form_table->RowCount - 1;

        label_9->Text = "TestMulti";
        this->form_table->Controls->Add(label_9, 0, _tfm);
    }
}

    for (int i = 1; i <= final; i++) {
        form_table->ColumnCount++;
        form_table->ColumnStyles->Add(gcnew
ColumnStyle(SizeType::Absolute, 50));

        label_0 = gcnew Label();
        label_0->Text = i.ToString();
        this->form_table->Controls->Add(label_0, i, 0);

        if (this->form_menustrip_view_data->Checked == true && i - 1 <
_buff.size()) {
            label_1 = gcnew Label();
            label_1->Text = myRound(_buff[i - 1], 2).ToString();
            this->form_table->Controls->Add(label_1, i, _d);
        }
        if (this->form_menustrip_view_forecast->Checked == true &&
start1 < i && i - start1 <= _buff_fm.size()) {
            if (this->form_menustrip_view_forecast_source->Checked ==
true) {
                label_2 = gcnew Label();
                label_2->Text = myRound(_buff_fs[i - start1 - 1],
2).ToString();

                this->form_table->Controls->Add(label_2, i, _fs);
            }
        }
    }
}

```

```

    }
    if (this->form_menustrip_view_forecast_approx3->Checked ==
true) {
        label_3 = gcnew Label();
        label_3->Text = myRound(_buff_fa3[i - start1 - 1],
2).ToString();
        this->form_table->Controls->Add(label_3, i, _fa3);
    }
    if (this->form_menustrip_view_forecast_approx5->Checked ==
true) {
        label_4 = gcnew Label();
        label_4->Text = myRound(_buff_fa5[i - start1 - 1],
2).ToString();
        this->form_table->Controls->Add(label_4, i, _fa5);
    }
    if (this->form_menustrip_view_forecast_multi->Checked ==
true) {
        label_5 = gcnew Label();
        label_5->Text = myRound(_buff_fm[i - start1 - 1],
2).ToString();
        this->form_table->Controls->Add(label_5, i, _fm);
    }
}
if (this->form_menustrip_view_forecastttest->Checked == true &&
start2 < i && i - start2 <= _buff_tfm.size()) {
    if (this->form_menustrip_view_forecastttest_source->Checked
== true) {
        label_6 = gcnew Label();
        label_6->Text = myRound(_buff_tfs[i - start2 - 1],
2).ToString();
        this->form_table->Controls->Add(label_6, i, _tfs);
    }
    if (this->form_menustrip_view_forecastttest_approx3-
>Checked == true) {
        label_7 = gcnew Label();
        label_7->Text = myRound(_buff_tfa3[i - start2 - 1],
2).ToString();
        this->form_table->Controls->Add(label_7, i, _tfa3);
    }
    if (this->form_menustrip_view_forecastttest_approx5-
>Checked == true) {
        label_8 = gcnew Label();

```

```

label_8->Text = myRound(_buff_tfa5[i - start2 - 1],
2).ToString();
        this->form_table->Controls->Add(label_8, i, _tfa5);
    }
    if (this->form_menustrip_view_forecasttest_multi->Checked
== true) {
        label_9 = gnew Label();
        label_9->Text = myRound(_buff_tfm[i - start2 - 1],
2).ToString();
        this->form_table->Controls->Add(label_9, i, _tfm);
    }
}
}
}

void Forecast::ForecastForm::UpdateStatusStrip() {
    vector<double> _kappa;
    _kappa = TestForecast.getKappaRange();
    this->form_statusstrip_label->Text = "Kappa [" +
_kappa[0].ToString() + ", " + _kappa[1].ToString() + "];"
}

void Forecast::ForecastForm::UnlockKappa() {
    this->form_menustrip_project_kappa->Enabled = true;
}

void Forecast::ForecastForm::UnlockForecast() {
    this->form_menustrip_project_forecast->Enabled = true;
}

void Forecast::ForecastForm::UnlockForecastTest() {
    this->form_menustrip_project_forecast_test->Enabled = true;
}

void Forecast::ForecastForm::UnlockDataView() {
    this->form_menustrip_view_data->Enabled = true;
}

void Forecast::ForecastForm::UnlockForecastView() {
    this->form_menustrip_view_forecast->Enabled = true;
    //Поставить автогалку
}

void Forecast::ForecastForm::UnlockForecastTestView() {
    this->form_menustrip_view_forecasttest->Enabled = true;
    //Поставить автогалку
}

```

```

System::Void
Forecast::ForecastForm::form_menustrip_file_create_Click(System::Object
^sender, System::EventArgs ^e) {
    TestForecast.defineDatasequence({ 23.33,33.72, 28.55, 28.02, 29.56,
26.53, 25.12, 24.73, 21.85, 24.31, 26.76, 30.42, 32.65, 30.96, 31.23 });
    UnlockKappa();
    return System::Void();
}

```

```

System::Void
Forecast::ForecastForm::form_menustrip_project_forecast_Click(System::Object
^sender, System::EventArgs ^e) {
    TestForecast.defineDataqeuenceRange();
    TestForecast.defineApproxScales();
    TestForecast.testForecast();
    TestForecast.defineForecast(4);

    ForecastStepForm ^ForecastStepV = gcnew ForecastStepForm();
    ForecastStepV->Show();

    UpdateGraph();
    UpdateTable();

    UnlockDataView();
    UnlockForecastView();
    UnlockForecastTestView();

    return System::Void();
}

```

```

System::Void
Forecast::ForecastForm::form_menustrip_project_kappa_manual_Click(System::Ob
ject ^sender, System::EventArgs ^e) {
    KappaForm ^KappaFormV = gcnew KappaForm();
    KappaFormV->Show();

    //while (secondForm) {
    //}

```

```

TestForecast.defineKappa(0.);
UpdateStatusStrip();
UnlockForecast();
return System::Void();
}

double myRound(double _value, int _after) {
    int _mul = 1;

    for (int i = 0; i < _after; i++) {
        _mul *= 10;
    }

    return (round(_value * double(_mul)) / double(_mul));
}

```

## Текст програми 2 – ForecastDS.h

```

#pragma once

#include <iostream>
#include <vector>

using namespace std;

class ForecastDS {
protected:
    struct struct_kappa_range {
        double low, medium, top;
        double range;
    };
    struct struct_datasequence_range {
        double min, max;
    };

    vector<double> datasequence;

    struct_kappa_range kappa;
    struct_datasequence_range datasequence_range;
    struct_datasequence_range approx_scale_range;

    vector<double> approx_scale_3;
    vector<double> approx_scale_5;

    vector<double> forecast_multi;
    vector<double> forecast_source;
    vector<double> forecast_approx3;
    vector<double> forecast_approx5;

    vector<double> test_forecast_multi;
    vector<double> test_forecast_source;
    vector<double> test_forecast_approx3;
    vector<double> test_forecast_approx5;
public:
    ForecastDS();

```

```

ForecastDS(vector<double>, double, double);
~ForecastDS();

void defineDatasequence(vector<double> _datasequence);
void defineKappa(double _value, double _range = 0.);
void defineKappa(double _value_low, double _value_top, bool _range);
void defineKappaAuto(double _value_low = -500., double _value_top =
500., double _step = 1., bool _set = true); //NEED REAL
void defineDatagequenceRange();
vector<double> defineApproxScale(double min, double max, int number);
void defineApproxScales();
void defineForecast(int _steps = 3);

double forecastMulti(double _m1, double _m2, double _m3, bool _save =
false);
double forecastSource(double _m1, double _m2, double _m3, bool _save =
false);
double forecastApprox3(double _m1, double _m2, double _m3, bool _save =
false);
double forecastApprox5(double _m1, double _m2, double _m3, bool _save =
false);

double forcastCalculateByMath(double _m1, double _m2, double _m3, double
_left = 0.5, double _right = 0.5);
double forcastCalculateByDed(double _m1, double _m2, double _m3, double
_left = 0.5, double _right = 0.5);

void testForcast();

void viewDatasequence();
void viewKappa();
void viewApproxScales();
void viewForecast(bool _test = true);

vector<double> getDatasequence();
vector<double> getForecastSource();
vector<double> getForecastApprox3();
vector<double> getForecastApprox5();
vector<double> getForecastMulti();
vector<double> getTestForecastSource();
vector<double> getTestForecastApprox3();
vector<double> getTestForecastApprox5();
vector<double> getTestForecastMulti();
vector<double> getKappaRange();

double interpol(double _value, double _range1_low, double _range1_top,
double _range2_low, double _range2_top);
int inRange(double _value, double _low, double _top);
double getApproxCenter(double _value, vector<double> _approx_scale);
double getDelta(double _m1, double _m2, double _m3);
double myRound(double _value, int _after = 0);
};

```

### Текст програми 3 – ForecastDS.cpp

```

#include <iostream>
#include <vector>
#include "ForecastDS.h"

using namespace std;

ForecastDS::ForecastDS() {}
ForecastDS::ForecastDS(vector<double> _datasequence, double _kappa, double
_percent) {

```

```

    defineDatasequence(_datasequence);
    defineKappa(_kappa, _percent);
}
ForecastDS::~ForecastDS() {}

void ForecastDS::defineDatasequence(vector<double> _datasequence) {
    datasequence = _datasequence;
}

void ForecastDS::defineKappa(double _value, double _range) {
    kappa.medium = _value;
    kappa.range = _range;

    kappa.low = kappa.medium * (1. - (kappa.range / 100.));
    kappa.top = kappa.medium * (1. + (kappa.range / 100.));
}

void ForecastDS::defineKappa(double _value_low, double _value_top, bool
_range) {
    kappa.low = _value_low;
    kappa.top = _value_top;

    kappa.medium = (kappa.low + kappa.top) / 2.;
    kappa.range = -1.;
}

void ForecastDS::defineKappaAuto(double _value_low, double _value_top, double
_step, bool _set) { //NEED REAL
}

void ForecastDS::defineDatasequenceRange() {
    for (int i = 0; i < datasequence.size(); i++) {
        if (i == 0) {
            datasequence_range.min = datasequence_range.max =
datasequence[i];
        }
        if (datasequence[i] < datasequence_range.min)
datasequence_range.min = datasequence[i];
        if (datasequence[i] > datasequence_range.max)
datasequence_range.max = datasequence[i];
    };

    approx_scale_range.min = datasequence_range.min * (1. - (2. / 100.));
    approx_scale_range.max = datasequence_range.max * (1. + (2. / 100.));
}

vector<double> ForecastDS::defineApproxScale(double _min, double _max, int
_number) {
    vector<double> ApproxScale;
    for (int i = 0; i <= _number; i++) {
        ApproxScale.push_back(interpol(double(i), 0., _number, _min,
_max));
    }
    return ApproxScale;
}

void ForecastDS::defineApproxScales() {
    approx_scale_3 = defineApproxScale(approx_scale_range.min,
approx_scale_range.max, 3);
    approx_scale_5 = defineApproxScale(approx_scale_range.min,
approx_scale_range.max, 5);
}

void ForecastDS::defineForecast(int _steps) {
    vector<double> _forecast;

    forecast_multi.clear();
    forecast_source.clear();
    forecast_approx3.clear();
    forecast_approx5.clear();
}

```

```

for (int i = 0; i < _steps; i++) {
    for (int j = 3; j >= 1; j--) {
        if (forecast_multi.size() >= j) {

            _forecast.push_back(forecast_multi[forecast_multi.size() - j]);
        }
        else {
            _forecast.push_back(datasequence[datasequence.size() -
(j - forecast_multi.size())]);
        }
    }

    forecastMulti(_forecast[_forecast.size() - 1],
_forecast[_forecast.size() - 2], _forecast[_forecast.size() - 3], true);
    _forecast.clear();
}

double ForecastDS::forecastMulti(double _m1, double _m2, double _m3, bool
_save) {
    double _ret;
    _ret = (forecastSource(_m1, _m2, _m3, _save) + forecastApprox3(_m1, _m2,
_m3, _save) + forecastApprox5(_m1, _m2, _m3, _save)) / 3.;
    if (_save == true) forecast_multi.push_back(_ret);
    return _ret;
}

double ForecastDS::forecastSource(double _m1, double _m2, double _m3, bool
_save) {
    double _trend, _ret;

    _trend = getDelta(_m1, _m2, _m3);

    switch (inRange(_trend, kappa.low, kappa.top)) {
    case -1:
        _ret = forcastCalculateByMath(_m1, _m3, _m2, 0.75, 0.25);
        break;
    case 0:
        _ret = forcastCalculateByMath(_m1, _m3, _m2, 0.5, 0.5);
        break;
    case 1:
        _ret = forcastCalculateByMath(_m1, _m3, _m2, 0.25, 0.75);
        break;
    }

    if (_save == true) forecast_source.push_back(_ret);
    return _ret;
}

double ForecastDS::forecastApprox3(double _m1, double _m2, double _m3, bool
_save) {
    double _trend, _ret;

    _trend = getDelta(_m1, _m2, _m3);
    _m1 = getApproxCenter(_m1, approx_scale_3);
    _m2 = getApproxCenter(_m2, approx_scale_3);
    _m3 = getApproxCenter(_m3, approx_scale_3);

    switch (inRange(_trend, kappa.low, kappa.top)) {
    case -1:
        _ret = forcastCalculateByMath(_m1, _m2, _m3, 0.75, 0.25);
        break;
    case 0:
        _ret = forcastCalculateByMath(_m1, _m2, _m3, 0.5, 0.5);
        break;
    }
}

```

```

case 1:
    _ret = forecastCalculateByMath(_m1, _m2, _m3, 0.25, 0.75);
    break;
}

if (_save == true) forecast_approx3.push_back(_ret);
return _ret;
}

double ForecastDS::forecastApprox5(double _m1, double _m2, double _m3, bool
_save) {
    double _trend, _ret;

    _trend = getDelta(_m1, _m2, _m3);
    _m1 = getApproxCenter(_m1, approx_scale_5);
    _m2 = getApproxCenter(_m2, approx_scale_5);
    _m3 = getApproxCenter(_m3, approx_scale_5);

    switch (inRange(_trend, kappa.low, kappa.top)) {
    case -1:
        _ret = forecastCalculateByMath(_m1, _m2, _m3, 0.75, 0.25);
        break;
    case 0:
        _ret = forecastCalculateByMath(_m1, _m2, _m3, 0.5, 0.5);
        break;
    case 1:
        _ret = forecastCalculateByMath(_m1, _m2, _m3, 0.25, 0.75);
        break;
    }

    if (_save == true) forecast_approx5.push_back(_ret);
    return _ret;
}

double ForecastDS::forecastCalculateByMath(double _m1, double _m2, double _m3,
double _left, double _right) {
    return 2. / ((_left / _m3) + (1. / _m2) + (_right / _m1));
}

double ForecastDS::forecastCalculateByDed(double _m1, double _m2, double _m3,
double _left, double _right) {
    return ((_left * _m3) + (1. * _m2) + (_right * _m1)) / 2.;
}

void ForecastDS::testForecast() {
    double _m1, _m2, _m3;

    test_forecast_multi.clear();
    test_forecast_source.clear();
    test_forecast_approx3.clear();
    test_forecast_approx5.clear();

    for (int i = 3; i < datasequence.size(); i++) {
        _m1 = datasequence[i - 1];
        _m2 = datasequence[i - 2];
        _m3 = datasequence[i - 3];

        test_forecast_source.push_back(forecastSource(_m1, _m2, _m3));
        test_forecast_approx3.push_back(forecastApprox3(_m1, _m2, _m3));
        test_forecast_approx5.push_back(forecastApprox5(_m1, _m2, _m3));

        test_forecast_multi.push_back((test_forecast_source[test_forecast_source
.size() - 1] + test_forecast_approx3[test_forecast_approx3.size() - 1] +
test_forecast_approx5[test_forecast_approx5.size() - 1]) / 3.);
    }
}

```

```

void ForecastDS::viewDatasequence() {
    cout << "Datasequence:";
    for (int i = 0; i < datasequence.size(); i++) {
        cout << "\t" << datasequence[i];
    }
    cout << endl;
}

void ForecastDS::viewKappa() {
    cout << "Kappa: [" << kappa.low << ", " << kappa.top << "]\n";
    cout << endl;
}

void ForecastDS::viewApproxScales() {
    cout << "Approx scale 3:";
    for (int i = 0; i < approx_scale_3.size(); i++) cout << "\t" <<
approx_scale_3[i];
    cout << endl;

    cout << "Approx scale 5:";
    for (int i = 0; i < approx_scale_5.size(); i++) cout << "\t" <<
approx_scale_5[i];
    cout << endl;
}

void ForecastDS::viewForcast(bool _test) {
    cout << "Forecast src:";
    if (_test == true) {
        cout << "\t\t\t";
        for (int i = 0; i < test_forecast_source.size(); i++) {
            cout << "\t" << myRound(test_forecast_source[i], 2);
        }
    }
    else {
        for (int i = 0; i < datasequence.size(); i++) {
            cout << "\t";
        }
    }
    for (int i = 0; i < forecast_source.size(); i++) {
        cout << "\t" << myRound(forecast_source[i], 2);
    }
    cout << endl;

    cout << "Forecast ar3:";
    if (_test == true) {
        cout << "\t\t\t";
        for (int i = 0; i < test_forecast_approx3.size(); i++) {
            cout << "\t" << myRound(test_forecast_approx3[i], 2);
        }
    }
    else {
        for (int i = 0; i < datasequence.size(); i++) {
            cout << "\t";
        }
    }
    for (int i = 0; i < forecast_approx3.size(); i++) {
        cout << "\t" << myRound(forecast_approx3[i], 2);
    }
    cout << endl;

    cout << "Forecast ar5:";
    if (_test == true) {
        cout << "\t\t\t";
        for (int i = 0; i < test_forecast_approx5.size(); i++) {
            cout << "\t" << myRound(test_forecast_approx5[i], 2);
        }
    }
}

```

```

    }
    else {
        for (int i = 0; i < datasequence.size(); i++) {
            cout << "\t";
        }
    }
    for (int i = 0; i < forecast_approx5.size(); i++) {
        cout << "\t" << myRound(forecast_approx5[i], 2);
    }
    cout << endl;

    cout << "Forecast mult:";
    if (_test == true) {
        cout << "\t\t\t";
        for (int i = 0; i < test_forecast_multi.size(); i++) {
            cout << "\t" << myRound(test_forecast_multi[i], 2);
        }
    }
    else {
        for (int i = 0; i < datasequence.size(); i++) {
            cout << "\t";
        }
    }
    for (int i = 0; i < forecast_multi.size(); i++) {
        cout << "\t" << myRound(forecast_multi[i], 2);
    }
    cout << endl;
}
vector<double> ForecastDS::getDatasequence() {
    return datasequence;
}
vector<double> ForecastDS::getForecastSource() {
    return forecast_source;
}
vector<double> ForecastDS::getForecastApprox3() {
    return forecast_approx3;
}
vector<double> ForecastDS::getForecastApprox5() {
    return forecast_approx5;
}
vector<double> ForecastDS::getForecastMulti() {
    return forecast_multi;
}
vector<double> ForecastDS::getTestForecastSource() {
    return test_forecast_source;
}
vector<double> ForecastDS::getTestForecastApprox3() {
    return test_forecast_approx3;
}
vector<double> ForecastDS::getTestForecastApprox5() {
    return test_forecast_approx5;
}
vector<double> ForecastDS::getTestForecastMulti() {
    return test_forecast_multi;
}
vector<double> ForecastDS::getKappaRange() {
    return vector<double>({kappa.low, kappa.top});
}
double ForecastDS::interpol(double _value, double _range1_low , double
_range1_top, double _range2_low, double _range2_top) {
    return (_range2_low + (((_value - _range1_low) / (_range1_top -
_range1_low)) * ((_range2_top - _range2_low) / 1.)));
}

```

```
int ForecastDS::inRange(double _value, double _low, double _top) {
    if (_value > _top) {
        return 1;
    }
    else {
        if (_value < _low) {
            return -1;
        }
        else {
            return 0;
        }
    }
}

double ForecastDS::getApproxCenter(double _value, vector<double>
_approx_scale) {
    for (int i = 0; i < _approx_scale.size() - 1; i++) {
        if (inRange(_value, _approx_scale[i], _approx_scale[i + 1]) == 0)
        {
            return (_approx_scale[i] + _approx_scale[i + 1]) / 2.;
        }
    }
    return 0.;
}

double ForecastDS::getDelta(double _m1, double _m2, double _m3) {
    return (_m1 - _m2) - (_m2 - _m3);
}

double ForecastDS::myRound(double _value, int _after) {
    int _mul = 1;

    for (int i = 0; i < _after; i++) {
        _mul *= 10;
    }

    return (round(_value * double(_mul)) / double(_mul));
}
```

61  
44165850.1218-01 12 01  
МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ЗАТВЕРДЖЕНО  
1116130.01165-01 12 01-ЛЗ

Дослідження методів моделювання часових послідовностей з  
нерівномірним кроком

Результати апробації результатів досліджень

1116130.01165-01 12 01

Аркушів 5

2024

## Результати апробації результатів досліджень



МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
МІНІСТЕРСТВО ІНФРАСТРУКТУРИ УКРАЇНИ  
УКРАЇНСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ НАУКИ ТА ТЕХНОЛОГІЙ  
СХІДНИЙ НАУКОВИЙ ЦЕНТР ТРАНСПОРТНОЇ АКАДЕМІЇ НАУК

ABSTRACTS  
OF THE XVII INTERNATIONAL CONFERENCE  
«MODERN INFORMATION AND COMMUNICATION  
TECHNOLOGIES ON A TRANSPORT, IN INDUSTRY AND  
EDUCATION»

13-14, December, 2023



СУЧАСНІ ІНФОРМАЦІЙНІ ТА  
КОМУНІКАЦІЙНІ  
ТЕХНОЛОГІЇ НА  
ТРАНСПОРТІ, В  
ПРОМИСЛОВІСТІ ТА ОСВІТІ

ТЕЗИ

XVII МІЖНАРОДНОЇ  
НАУКОВО-  
ПРАКТИЧНОЇ  
КОНФЕРЕНЦІЇ  
13-14 ГРУДНЯ 2023

ДНІПРО  
2023

**ЗМІСТ (частина)**

Дослідження методів та засобів аутентифікації користувача у реальному часі шляхом розпізнавання обличчя..... **Помилка! Закладку не визначено.**

Різниченко М. О., Горбова О. В., Український державний університет науки і технологій, Україна

Поведінкові стратегії Інтернет бізнесу ..... **Помилка! Закладку не визначено.**

Савчук Л. М., Ковальчук Є. В. Український державний університет науки і технологій, Інститут промислових та бізнес технологій, Україна

Вибір цілі, критерію і норми при проектуванні інтелектуальної системи управління організаційно-технічними процесами..... **Помилка! Закладку не визначено.**

Самойлов С.П., Український державний університет науки і технологій, Україна

Розпізнавання рукописних символів за допомогою нейронної мережі..... **Помилка! Закладку не визначено.**

Середа О.А., Горячкін В.М., Український державний університет науки і технологій

Підвищення якості коду мобільних застосунків методами рефакторингу з використанням генеративного штучного інтелекту ..... **Помилка! Закладку не визначено.**

Сирота О. А., Горячкін В.М., Український державний університет науки і технологій

Багатовимірні нечіткі моделі моніторингу і реабілітації з нерівномірним інтервалом спостережень **Помилка! Закладку не визначено.**

Скалозуб В.В., Горячкін В.М., Клименко І.В., Мурашов О.В., Український державний університет науки і технологій

Процедури редукції і капша статистики у завданнях формування достовірних математичних моделей класифікації невизначених даних **Помилка! Закладку не визначено.**

Скалозуб В.В., Терлецький І.А., Дудник І.О., Український державний університет науки і технологій, Скалозуб М. РауPal, Швеція

Порівняльний аналіз комп'ютерних моделей прогнозування часових послідовностей з нерівномірним і нечітким кроком спостережень..... **Помилка! Закладку не визначено.**

Скалозуб В.В., Медич С.О., Старина А.К., Український державний університет науки і технологій, Україна

Розвиток і дослідження методів моделювання та прогнозування часових послідовностей з нерівномірним кроком і нечіткими параметрами спостережень **Помилка! Закладку не визначено.**

Скалозуб В.В., Медич С.О., Старина А.К., Український державний університет науки і технологій



Публікація ТЕЗИ XVII Міжнародної науково-практичної конференції «СУЧАСНІ ІНФОРМАЦІЙНІ ТА КОМУНІКАЦІЙНІ ТЕХНОЛОГІЇ НА ТРАНСПОРТІ, В ПРОМИСЛОВІСТІ ТА ОСВІТІ. 13.12.2023 – 14.12.2023, Український державний університет науки і технологій, Україна, стор. 85

**Порівняльний аналіз комп'ютерних моделей прогнозування часових послідовностей з нерівномірним і нечітким кроком спостережень**

Скалозуб В.В., Медич С.О., Старина А.К.

Український державний університет науки і технологій, Україна

У доповіді приведені результати дослідження математичних і комп'ютерних моделей, призначених для аналізу та прогнозування послідовностей спостережень над процесами, що мають нерівномірні та нечіткі інтервали (ННІ) між контрольними вимірюваннями. При цьому були розглянуті три види математичних моделей зазначених процесів: - сепарабельні моделі (СПМ), в яких послідовності нерівномірних інтервалів та рівнів процесів моделюються окремо, з наступним агрегуванням; - нечітко-реляційна модель (НРМ), призначена для аналізу та прогнозування процесів типу ННІ. Досліджено можливості НРМ моделей першого та другого порядку, що залежить від кількості попередніх етапів, які ураховуються при моделюванні та прогнозуванні; - комбінованих/комплексних багатовимірних нечітких моделей (CFTS). В основу методики та реалізацію моделей CFTS покладено визначені власні ознаки зазначених процесів спостережень та отримання первинних даних. Відмінність указаних вище математичних і комп'ютерних моделей процесів з ННІ полягає в різних формах урахування попередньої інформації, яка використана при постановках завдань дослідження.

У сепарабельних моделях СПМ було введено припущення, передумова формування алгоритмів аналізу та прогнозування, що кожна складова моделі, в тому числі часова координата, можуть досліджуватися незалежно між собою. Системну єдність процесу забезпечує координата часу, за якою об'єднуються всі інші. У нечітко-реляційних моделях НРМ в області можливих значень інтервалів між спостереженнями водиться апроксимація системою нечітких величин (НВ). За допомогою такої апроксимації виконується перехід від довільних форм нечітких інтервалів до їх визначених моделей (термів). Далі за допомогою уведення  $\alpha$ -рівнів НВ виконується перехід до моделей процесів з нерівномірними послідовностями інтервалів спостережень. Такі  $\alpha$ -рівневі послідовності дійсних процесів ННІ реалізуються алгоритмами СПМ. За рахунок  $\alpha$ -рівневих числових послідовностей визначається прогнозне значення наступного інтервалу для окремих рівнів. Прогнозоване значення величини інтервалу в цілому на наступному кроці процесу розраховується за методом скаляризації «центр ваги».

Відмінність моделей CFTS визначається тим, що в них урахована природа виникнення нерівномірних інтервалів спостережень. При тому рахують що інтервали, як інші параметри моделі, відображають системну єдність саме контрольованого процесу. Проводиться границя між «регламентом» отримання даних спостережень (за природою процесів), а також формою представлення даних спостережень. CFTS узагальнюють моделі FTS багатовимірних часових рядів порядку «n» з «m» вхідними і одним результуючим параметром. Розвиток процедур у CFTS визначається тим, що різні параметри процесів можуть мати різний порядок передісторії «n» (кількість попередніх значень), а також вони можуть вимірюватися різними типами даних та формами відображення оцінок невизначеності. Суттєва змістовна відмінність CFTS визначається тим, що вони забезпечують певну форму представлення результатів моделювання - прогнозування періодів до *наступних* контрольних заходів та очікувані оцінки показників спостережень. Прогнозування процесів з нерівномірними інтервалами спостережень в будь який момент часу ними не передбачається. Тож при застосуванні моделей CFTS необхідно саме так формулювати завдання дослідження та прогнозування, зокрема.

У доповіді приведені програмні засоби та реалізації комп'ютерних моделей виду СПМ, НРМ та CFTS, які забезпечують аналіз і прогнозування мульти-параметричних процесів моніторингу та реабілітації хворих на діабет для даних у формі часових послідовностей з нерівномірним і нечітким кроком спостережень.

**Розвиток і дослідження методів моделювання та прогнозування часових послідовностей з нерівномірним кроком і нечіткими параметрами спостережень**

Скалозуб В.В., Медич С.О., Старина А.К.  
Український державний університет науки і технологій

Проблеми комп'ютерного моделювання недетермінованих процесів з нечіткими параметрами, а також нерівномірними інтервалами спостережень (НПНІ), привертають все більше уваги, що показує аналіз публікацій, а також суттєву обмеженість підходів до їх моделювання. В ряді досліджень зроблено висновок, що як правило, інтерполяція нерівномірного ряду для отримання послідовності з постійним інтервалом дискретизації змінює його властивості. Тому часто перевагу віддають підходам, які працюють з нерівномірними даними безпосередньо, уникаючи необхідності переходу до явного постійного кроку інтерполяції. Для моделювання процесів НПНІ нами були запропоновані сепарабельні моделі (СПМ), в яких послідовності нерівномірних інтервалів та рівнів процесів моделюються окремо, з наступним агрегуванням. Важливим змістовним завданням щодо організації та моделювання нечітких процесів упорядкування являється завдання моніторингу процесів реабілітації захворювань, а також діагностування і прогнозування станів складних систем. Наприклад, фактично нерівномірність кроку між контрольними заходами хворих, а також оцінювані діагностичні показники являються нечіткими. З точки зору практики метою використання результатів моделювання виступає оцінка періоду до наступної контрольованої події (відвідування лікаря, також і рівень його оцінюваних параметрів).

В доповіді представлені результати із розвитку та дослідження методів моделювання та прогнозування часових послідовностей з нерівномірним кроком і нечіткими параметрами спостережень. В якості першоджерел даних НПНІ використані дані клінічного моніторингу процесів лікування. Для забезпечення можливості моделювання таких коротких послідовностей (клінічних моніторингу *конкретного* хворого) в дослідженнях спочатку дані спостережень розмножуються методом бутстреп, що дозволило отримати вибірки більших розмірів при збереженні властивостей процесів.

З метою моделювання процесів НПНІ застосовується СПМ, яка отримала певний розвиток. А саме, в роботі запропонована нова, нечітко-реляційна модель (НРМ), призначена для аналізу та прогнозування процесів типу НПНІ. Досліджено можливості НРМ моделей першого та другого порядку, що залежить від кількості попередніх етапів, які ураховуються при моделюванні та прогнозуванні. Для моделювання процесів з нечітким кроком виконуються процедури переходу від невідомих моделей інтервалів до нечіткого модельного представлення цих послідовностей. Для цього в області можливих значень інтервалів водиться моделююча система нечітких величин (НВ), через яку виконується перехід від довільних форм інтервалів до їх моделей, що є апроксимацією. Наступним кроком моделювання є перетворення за допомогою введення  $\alpha$ -рівнів НВ послідовностей апроксимації в нерівномірні послідовності спостережень. Отримані  $\alpha$ -рівневі нерівномірні послідовності дійсних процесів НПНІ реалізуються засобами алгоритмів СПМ. При цьому шляхом моделювання числових послідовностей визначається прогнозне значення наступного інтервалу для кожного  $\alpha$ -рівня. Для визначення остаточної оцінки інтервалу (показника) на наступному кроці виконується процедура скаляризації методом центру ваги. В роботі також удосконалено процедуру формування комбінованих алгоритмів моделювання процесів НПНІ, що забезпечує коректність результатів комп'ютерного моделювання та підвищення точності показників процесів.

У роботі приводяться структура програми дослідження та прогнозування процесів НПНІ, приклади реалізації завдань моделювання, які дозволяють оцінити достовірність та ефективність запропонованих методів моделювання та прогнозування часових послідовностей з нерівномірним кроком та нечіткими параметрами спостережень.